## REMARKS

Claims 1-20 remain pending in the application. Claims 1-20 stand rejected, with claim 18 additionally objected to. Claim 18 is amended in response to the Examiner's rejection and objections. Additionally, claims 4, 11, 13, 14, 17 and 19 are amended for editorial clarity. No new matter is added to the claims with these amendments.

### 1.    Claim Objections

The Examiner objected to claim 18, specifically stating that claim 18 is not in the form of a single sentence and contains obvious typographical errors such as duplicate text. Accordingly, Applicants have amended claim 18 such that the claim reads as a complete sentence, pursuant to Patent Rule §1.75, and additionally to correct typographical errors and eliminate duplicate text. Given these amendments and the arguments laid out herein below, Applicants respectfully request withdrawal of the objection to and rejection of claim 18, and allowance of the claim.

### 2.    Claim Rejections – 35 USC § 103

The following is a quotation from the MPEP setting forth the three basic criteria that must be met to establish a *prima facie* case of obviousness.

> To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the references or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. MPEP, §2142, citing *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

**3.**    Claims 1-20 stand rejected under 35 USC §103 over CyberCop Scanner by Network Associates as described in the Info World article entitled "Test Center Comparison" (hereinafter, "CyberCop") in view of Info World article entitled "The Ins and Outs of a Network Security Audit" (hereinafter, "Security Audit"). Applicants respectfully disagree and traverse the rejection since, among other reasons, CyberCop is not prior art to the inventions of claims 1-20.

In the Office Action of 11/05/2003, the Examiner states that "CyberCop discloses the instant invention essentially as claimed with the exception that CyberCop does not specify generating a configuration baseline or a file system database for use in other utility functions"(page 2), but that "Security Audit discloses... that network audit results should be stored for comparison to future audits"(page 3). The Examiner therefore contends that claims 1-20 are unpatentable over CyberCop in view of Security Audit. However, CyberCop, with an effective date of February 8, 1999, is not prior art to the present application, which is a Continuation of U.S. Serial No. 09/333,547, which claims priority of provisional application number 60/091,270, filed 15 June 1998 (hereinafter, "the provisional application"). Support for each of claims 1-20 is found throughout the provisional application, including but not limited to those locations detailed herein below.

Provisional application no. 60/091,270 (referenced herein) was filed without page numbering. Therefore, for the Examiner's convenience, Applicants have applied page numbering to the provisional, and submit herewith an Appendix A consisting of those pages cited in this Response.

## Support for Independent Claims

<u>Claim 1</u>

The elements of claim 1 (shown in italics and within quotation marks) are supported in the following sections of the provisional application, among others:

- *"a security system for a computer apparatus, wherein said computer apparatus includes a processor and system memory"* is supported, for example, in the first paragraph on page 44 (page entitled "DMW Introduces HostCHECK for UNIX Advanced Security Tool Set"). Such a UNIX system has a processor and system memory. Further support is found throughout page 65 (page entitled "...to protect: HostCHECK powers your company for e-business with a suite of nine security tools integrated into one easy-to-use program").
- the security system comprising *"at least one security module which under direction from the processor accesses and analyzes selected portions of the computer apparatus to identify vulnerabilities"* is supported in the second and third paragraphs on page 45 (page entitled "DMW Introduces HostCHECK

(Page 2 of 3))", and from the last paragraph of page 51 to the end of page 52, in the section entitled "Nine Security Modules".

- the security system comprising *"at least one utility module which under the direction from the processor, performs various utility functions with regards to the computer apparatus in response to the identified vulnerabilities"* is supported on page 63, third paragraph; in the first paragraph of the first column of page 67,in the pie chart found on page 70 of the provisional application, and in the first paragraph on page 110, under the subheading "Highest Level Component Interaction"

- the security system comprising *"a security system memory which contains security information for performing the analysis of the computer apparatus"* is supported, for example, on page 63, in the paragraph entitled "Immediate Security Improvements", and in the description of the DMW Vulnerability Database found on page 97.

## Claim 11

Support for the elements of claim 11 (shown in italics and within quotation marks) may be found in the following sections of the provisional application, among others.

- *"A method of providing a security assessment for a computer system which includes a system memory"* is supported throughout the provisional application, for example in the last paragraph on page 44 through the final bullet point, page 45, and throughout page 65 (page entitled "…to protect: HostCHECK powers your company for e-business with a suite of nine security tools integrated into one easy-to-use program").

- the step of *"providing a security subsystem in the computer system such that functionality of the security subsystem is directed through a processor for the computer system"* is supported throughout the provisional application, for example, in the figures shown on pages 110, 112, 115, 116, and on pages 117-118, in the paragraphs following the heading "Subsystem Diagrams".

- the security performing the step of *"identifying a configuration of the system"* is supported in first paragraph (entitled "Configuration Check") of the second column on page 65, in the first bullet point on page 70, and on page 111 (also

identified as page 13 of the Functional Description for HOSTCHECK 2.0), under the heading "Configuration Detection System".

- the security performing the step of *"accessing the system memory"* is supported at least in the paragraphs in the subsection entitled "Directory Check Locates Security Flaws and Prompts Auto Correction", page 53, and on page 111, under the subheading "Directory Scanner".

- the security further *"performing at least one procedure to provide a security assessment for at least one aspect of the computer system"* is supported in the second and third paragraphs on page 45 (page entitled "DMW Introduces HostCHECK (Page 2 of 3))"; from the last paragraph of page 51 to the end of page 52, in the section entitled "Nine Security Modules", and in the second paragraph under the subheading :Highest Level Component Interaction", page 110.

- *"as a result of any vulnerabilities discovered in the assessment, identifying corrective measures to be taken with regards to the computer system"* as a result of vulnerabilities discovered is supported in the third bullet point on page 45 and in the third bullet point on page 76 of the provisional application.

- *"reporting the discovered vulnerability and the identified corrective measures"* is supported, for example, by the third through sixth bullet points on page 89; in the last two paragraphs on page 45, under "Intelligent Monitoring and Reporting", and in the section entitled "Usable Reports", middle of page 51.

- support for the security performing a step comprising *"upon receiving an appropriate command, initiating the corrective measures"* is found on page 49 and in the last paragraph on page 63 (under "Immediate Security Improvements") of the provisional application

### Support for Dependent Claims

Claims 2-10 and 12-20 depend from independent base claims 1 and 11 (respectively), and therefore receive like benefit of support from the provisional application. The provisional further supports the features of dependent claims 2-10 and 11-20 (shown in italics and within quotation marks) as exemplified below:

Claim 2

Support for the security system *"including at least one graphical user interface in connection with the computer apparatus through which a system user may direct operations of the security system"*, is shown in the screens exemplified throughout the provisional, for example, on pages 48, 49, 74, 93 and 95, and further on page 51, second paragraph (Section entitled "A Friendly GUI").

Claim 3

Support for the security system *"including a reporting module which provides status information to the GUI with regards to operations of the security system"* is found in the paragraph on page 127, and in the paragraph entitled "A Friendly GUI" on page 51 of the provisional application.

Claim 4

Support for the security modules including at least one of:

- *"a configuration/system module which performs an initial analysis of the computer system to acquire configuration information"* is found in the first paragraph on page 45, and in first paragraph (entitled "Configuration Check") of the second column on page 65, and in the six paragraphs under the subheading "Configuration Detection Systems", page 111;

- *"a directory checking module which analyzes directories and files in the system memory to determine if security critical files have been tampered with"* is found in the description of "Directory Check" on pages 53; 65 (first column), and 73-74, and in the description of "Directory Scanner", page 112;

- *"a user manager module which analyzes the system memory with regards to improper or invalid permissions given to users of the system for accessing particular files"* is shown in the descriptions of "User Manager" and "User Check", covered in the first and second paragraphs on page 56, respectively, and further under the subsection "User Manager", page 115;

- *"an integrity checking module which analyzes files in the system memory to identify system vulnerabilities"* is found in the descriptions of "Integrity Check", page 55, and "Integrity Checker", page 114;

- *"a network checking module which analyzes the computer apparatus to identify vulnerabilities created as a result of the computer apparatus*

*connecting with a data network"* is found, for example, in the description of "Network Check" on page 59, first paragraph, and in the first paragraph on page 116, under the subheading "Network Manager"; and

- *"a password checking module which analyzes passwords for users of the computer apparatus to identify vulnerabilities"* is found from the final paragraph on page 56 through the sixth paragraph on page 57, and under "Integrity Checker", page 114 of the provisional application

## Claim 5

The utility modules including at least one of:

- a *" user manager module including functionality to create a user account, modify the user account, delete the user account, create a user template, edit the user template, and delete the user template"* is supported on page 106 in the bullet points found under the subheading "User Management Functions"; page 115, under the subheading "User Manager"; and in the "Appendix A-- Alphabetized Function List" beginning on page 216, specifically, in the fourth and 33$^{rd}$ functions on page 216, and in the third function on page 217. Additional support is found on page 228 of the application (corresponding to page B-8 of the "Appendix B-- Directory Structure Contents").
- *"a file removal module which deletes selected files from the system memory and removes links to the deleted file"* is supported on page 57, last paragraph entitled "RemoveIT", and on pages 86 and 87 of the provisional application.
- *"a file marking module which marks selected files"* is supported in the description and screen printout of the "MarkIT" feature, found on pages 92 and 93 of the provisional application.
- *"a scheduling module which may be employed to schedule any and all of the security modules to perform analysis of the system memory"* is supported on pages 94-95 and on page 59 of the provisional application, in the third paragraph describing "Schedule IT".

## Claim 6

*"...the computer apparatus comprising a Unix server"* is supported on pages 44-48 of the provisional application, in the article entitled "DMW Introduces HostCHECK for UNIX Advanced Security Tool Set".

Claim 7

"*The security system of claim 6 wherein the server is connected to a data network*" is supported at least on page 60, third paragraph, and on page 63, final paragraph.

Claim 8

"*a plurality of interface screens*" being "*presented at the GUI for controlling operations of the security system*" is found at page 51, second paragraph; on page 127, and in the depiction of various interface screens on pages 48, 49, 74, 93 and 95.

Claim 9

Support for the system memory comprising "*a list of known vulnerabilities which may be employed by the integrity checking module*" is found at least on page 55, final paragraph, and in the diagram shown on page 115 of the provisional application.

Claim 10

Support for the system memory comprising "*dictionaries and other tools employed by the password checking module*" is found in the third bullet point describing "CrackIT" on page 83, and in the final bullet point under "Password Cracker" on page 113 of the provisional application.

Claim 12

The elements of claim 12 are supported throughout the provisional application as exemplified below:

- support for "*performing an analysis of the directories and files in the system memory to determine if security critical files have been tampered with*" is found on page 58, final two paragraphs; in the first paragraph, first column of page 65; in the third bullet point describing "Directory Check" on page 73, and under the section "Directory Scanner" on page 112 of the provisional application.

- "*analyzing the system memory with regards to improper or invalid permission given to users of the system for accessing particular file*" is supported, for example, in the first and fourth bullet points of the description of "Directory Check", page 7; in the description of "Permissions Check", page 75, and under "User Manger", page 115.

- support for *"analyzing the system memory to identify system vulnerabilities"* is found throughout the provisional, for example, in the final paragraph on page 44, under the subheading "Elements of Improved Security", and on page 114 under "Integrity Checker".

- *"analyzing the computer apparatus to identify vulnerabilities created as a result of the computer apparatus connecting to a data network"* is supported in the description of "Network Check" on page 59, first paragraph, and in the first paragraph on page 116, under the subheading "Network Manager".

- support for *"analyzing passwords for users of the computer apparatus to identify vulnerabilities"* is found in the first through sixth paragraphs on page 57, and under "Password Cracker, page 113 of the provisional application.

Claim 13

Support for the elements of claim 13 are found at least in the sections of the provisional application identified below:

- support for *"amending, deleting or creating user account"* based on identified vulnerabilities is found on page 106 in the bullet points found under the subheading "User Management Functions"; on page 115, under the subheading "User Manager"; and in the "Appendix A--Alphabetized Function List" beginning on page 216, specifically, in the fourth and 33$^{rd}$ functions on page 216, and in the third function on page 217.

- support for *"amending, deleting, or creating user templates"* is found in the sections cited in the above bullet point, and additionally on page 228 of the application (which corresponds to page B-8 of the "Appendix B-- Directory Structure Contents").

- *"deleting selected files from the system memory and removing links to said file"* is supported in the descriptions of "RemoveIT" (page 57), "WipeIT" (page 116), and "Secure File Wipe" (pages 125-126).

- *"marking of selected files within the system memory"* is supported, for example, in the section entitled "Mark/Unmark Subsystem", found on page 187 of the provisional application.

Claim 14

Support for the steps of *"accessing individual files in the system memory"*; *"identifying the type of file contained therein"*; and *"making a determination as to whether the permissions for the identified file are secure"* are supported under the subheading "Directory Check", found in the first column on page 65, and under "Directory Scanner", page 112.

Support for the step of *"providing a report describing the insecurity"* is supported in the bullet point description of "AutoCorrect" on page 76, and in the bullet point description of "ReportIT" on page 89, and in the first paragraph on page 115 of the provisional application.

Support for the steps of *"providing corrections for the detected files which are insecure"* and *"initializing corrective action upon receiving direction"* are supported in the paragraph on page 49; in the paragraph on page 54; in the bullet point description of "AutoCorrect" on page 76, and in the first paragraph on page 113.

Claim 15

Support for the steps of *"performing a user check to see if a user owns his or her home directory"*; *"performing a check to see if the user's group owns the home directory"*; *"performing a check to see if user related files are valid"*; and *"performing a check to see if the user's directory exists"* is found, for example, on page 106 under the section entitled "User Security Examinations", particularly in the lead paragraph and in the first, second and third bullet points.

Claim 16

The step of *"providing a vulnerability database which includes a number of identified system vulnerabilities"* is supported at least on page 107, under the subsection entitled "Vulnerability Database".

The steps of *"accessing the individual files in the system memory"*; *"determining whether the file's owner matches a predetermined profile"*; *"determining whether the file's group matches a predetermined profile"*; and *"determining whether the permissions associated with the file match a predetermined profile"*; are supported in the section cited immediately above, and further supported on pages 135-136, in the sections entitled "Profile Database" and "Vulnerability Database"

The step of *"determining whether the files predate a patch"* is supported at least on page 107, in the first bullet point under the subsection entitled "Vulnerability Test Methods" under the subsection entitled "Vulnerability Database".

The step of *"providing a report on any vulnerabilities which may exist in the system memory"* is supported in the bullet point description of "AutoCorrect" on page 76, and in the bullet point description of "ReportIT" on page 89.

## Claim 17

The steps of *"checking for insecure configuration files"*; *"checking running of excessive system services"*; and *"checking whether the computer system is running in the promiscuous mode"* are supported at least on page 116, in the section under the heading entitled "Network Manager"

## Claim 18

The following steps of claim 18 are supported on page 124, in steps 1-17 under "Password Checker" (see especially items 1, 2, 7-9 and 12). Additional support is found on page 57, paragraphs 1-6 and on page 105, Phases 1-9, of the provisional application.

- *"identifying all passwords for the users of the computer system"*;
- *"reading the passwords and for each identifying a next similar salt entry"*;
- *"identifying a next predetermined number of words from the dictionary"*;
- *"performing a word filtering method with regards to the passwords to add to the word list"*;
- *"determining whether the word is in the list"*; and
- *"removing the user from the list."*

## Claim 19

Support for the method of providing a security assessment for a computer system including the step of *"displaying a result of the security analysis via a graphical user interface"* is found in the screens exemplified throughout the provisional, for example, on pages 48, 49, 74, 93 and 95, and further in the description of "A Friendly GUI", page 51.

## Claim 20

Support for the method of providing a security assessment for a computer system *"wherein the computer system is connected to a data network"* is supported at least on page 60, third paragraph; on page 63, final paragraph, and in the diagram on page 128.

As detailed above, claims 1-20 are supported by provisional application number 60/091,270, filed 15 June 1998. CyberCop, with an effective date of February 8, 1999, is therefore not prior art to the present application. Additionally, CyberCop fails to teach each and every element of claims 1-20. It is clear from the Examiner's comments that Security Audit also fails to teach each and every element of claims 1-20, as required under 35 U.S.C. §103, since, for example, Security Audit must now teach every element without benefit of CyberCop. Note the following quotation of from the MPEP setting forth the three basic criteria that must be met to establish a *prima facie* case of obviousness:

> To establish a *prima facie* case of obviousness, three basic criteria must be met. First, there must be some suggestion or motivation, either in the references themselves or in the knowledge generally available to one of ordinary skill in the art, to modify the references or to combine reference teachings. Second, there must be a reasonable expectation of success. Finally, the prior art reference (or references when combined) must teach or suggest all the claim limitations. MPEP, §2142, citing *In re Vaeck*, 947 F.2d 488, 20 USPQ2d 1438 (Fed. Cir. 1991).

Security Audit cannot teach each element of claims 1-20 and, therefore, fails under these requirements. Reconsideration and allowance of claims 1-20 is now requested.

According to MPEP 715, an inventor affidavit under 37 C.F.R. §1.131 is inappropriate in this circumstance, since the effective filing date of the present application (claims 1-20) is prior to the earliest priority date (February 8, 1999) of CyberCop.

## CONCLUSION

In summary, Applicants have shown support for all of Claims 1-20 within the provisional application. As the provisional application was filed prior to the CyberCop effective date, CyberCop is not available as prior art to claims 1-20. Therefore, withdrawal of all of the Examiner's rejections under 35 USC §103 is requested. Applicants have amended Claim 18 to overcome the Examiner's rejection and objections, and have further amended claims 4, 11, 13, 14, 17 and 19 for editorial clarity.

In view of the above Amendments and Remarks, Applicants have addressed all issues raised in the Office Action dated November 5, 2003, and respectfully solicit a Notice of Allowance for claims 1-20. Should any issues remain, the Examiner is encouraged to telephone the undersigned attorney.

A Petition for one month's extension of time to reply is submitted herewith, extending the period for reply up to and including March 5, 2004. Authorization to charge the necessary fee of $55 for a small entity to Deposit Account 12-0600 is granted within the attached Petition for one month's extension of time. It is believed that no further fees are due; however, if any additional fee is required in connection with this Amendment and Response, the Commissioner is further authorized to charge such fee to Deposit Account 12-0600.

Respectfully submitted,

Date: **Mar 1st, 2004**          By: _____
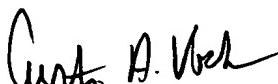
Curtis A. Vock, Reg. No. 38,356
LATHROP & GAGE L.C.
4845 Pearl East Circle, Suite 300
Boulder, CO 80301
Telephone: (720) 931-3011
Facsimile: (720) 931-3001

Response to Office Action mailed 11/5/2003 in U.S. Serial No. 09/834,334, entitled
METHOD AND APPARATUS FOR ASSESSING THE SECURITY OF A COMPUTER SYSTEM

# APPENDIX A

to

Response to Office Action mailed 11/5/2003 in U.S. Serial No. 09/834,334,

entitled

**METHOD AND APPARATUS FOR ASSESSING THE SECURITY OF A COMPUTER SYSTEM**

## DMW Introduces HostCHECK for UNIX Advanced Security Tool Set

### Powerful Solution Brings Security Expertise In-House, Enables E-Business

Colorado Springs, CO – April 28, 1998 – DMW Worldwide LLC today unveiled HostCHECK™ for UNIX, an integrated suite of tools that delivers immediate enterprise security improvements and allows users to bring security expertise in-house. Featuring a JAVA-based graphical interface and one of the industry's largest vulnerability databases that contains almost 2,000 validated exploits, HostCHECK provides a simple, yet powerful way for companies to maintain a high level security posture on an ongoing basis. The product set is part of DMW's family of adaptive e-business solutions that helps companies leverage the power of advanced networks in order to conduct business and deliver service online in a secure environment.

"Even though information protection has become one of the greatest concerns for CIOs today, most companies still rely only on firewall devices and also lack the tools and expertise in-house to ensure that all systems are safe from intrusion and suspicious behavior," said Dr. Bruce Hartley, executive vice president and chief technology officer for DMW. "HostCHECK is a key element of an enterprise security architecture that helps companies realize an immediate improvement in security, then easily maintain that level of security as the company changes."

### Elements of Improved Security

HostCHECK features nine modules that each focus on a specific security aspect of a UNIX-based host machine. Working together, these tools determine the vulnerabilities and weaknesses of each system, giving users the ability to evaluate, assess, correct, manage and monitor security.

— more —

44.

(Submitted in Appendix A of the

- Evaluate — The Configuration Check module automatically detects system characteristics, allowing HostCHECK to correctly configure itself on the installed platform. It identifies files considered critical to the security configuration of the computer or files that, if tampered with, can be used for penetration purposes.

- Assess — Several modules perform a thorough security assessment on the host system. These include Directory Check, Integrity Check, Network Check and User Check. DMW's CrackIT is another assessment module capable of cracking passwords locally or remotely with the goal of helping organizations improve the integrity of file access passwords. Directory Check, which · searches files for insecure permissions, is extremely fast, checking the security of up to 30,000 files in 60 seconds.

- Correct and Improve — The AutoCorrect feature is designed so that when HostCHECK detects a problem, it displays a screen prompt that identifies the vulnerability, suggests a correction and provides the rationale and impact of that correction. This feature provides the host with an immediate improvement in security. Other modules that help correct and improve host security include MarkIT, allowing users to mark security critical files; ReviewIT, which allows users to review and change audit trail settings; RemoveIT, a secure file wipe program; and the User Manager.

- Manage Configuration — The User Manager provides administrators with a standard interface that can be run across multiple UNIX platforms and allows them to manage account and group structures uniformly. This eliminates the creation of new security vulnerabilities when new accounts are created, thus greatly improving system administration and reducing security weaknesses.

- Protect and Monitor — The ReviewIT feature allows the administrator to review and modify audit trail settings, while Network Check searches for sniffers running on the host and for insecure entries in the network configuration.

### Intelligent Monitoring and Reporting

HostCHECK's ScheduleIT utility allows users to run checks on a regular basis, specific to the needs of each business, thus monitoring and protecting the security posture on an ongoing basis.

While most security tools require an in-depth knowledge of security to understand their results, HostCHECK generates reports in simple formats that make the nature of each security problem easy to understand. The user can select comprehensive or specific reports, which can be printed, stored or shared via email to a specific user account, such as a manager. All reports are archived so that comparisons can be made between new and previous reports, making an intrusion simple to detect.

— more —

45.

## Pricing and Availability

HostCHECK will be available from June through August at the initial price of $199 (U.S.) for a single host. After that time, the list price will be $295 per host. Multi-host, site license and unlimited host pricing is also available. DMW will provide software maintenance packages, which include software updates, vulnerability database updates and technical support. HostCHECK for UNIX was developed in ANSI C to optimize speed and portability, requires 32 Mb of RAM and currently supports Solaris 2.3 or higher, SunOS 4.1.X, FreeBSD, HP-UX v9 and v10, Digital UNIX 2.X, Irix 6.X, AIX 4.X and Linux 2.X. DMW plans to introduce a version for the NT platform later this year.

## About DMW

DMW Worldwide LLC develops electronic business solutions that adapt quickly to changing technology and customer demands. The company's Timarou™ family of integrated solutions provides proactive customer care, convergent billing and network care. By combining these products with strategic business and network services, clients receive a seamless suite of solutions to help them efficiently and securely conduct business over the network, as well as proactively manage user and customer needs. Timarou solutions are flexible, adapting to business environments and redefining what is possible with e-business — exceptional customer relationships, real-time billing, end-to-end information services management and secure worldwide communications. The privately held company is headquartered in Colorado Springs, with additional offices in Boulder, New York, Silicon Valley, London and Tokyo. For more information, contact (800) 369-4768 or visit www.dmwworldwide.com on the web.

DMW will feature HostCHECK demonstrations at the Networld+Interop conference, May 5-7 in the Las Vegas Convention Center, booth #5508 in the South Hall. Contact Susan MacCall to schedule press appointments.

###

DMW, HostCHECK and Timarou are trademarks of DMW Worldwide LLC. Other product names may be trademarks of their respective owners.

# DMW

**DMW WORLDWIDE**

4965 North 30th Street • Colorado Springs, CO 80919
Tel: 719-548-1101 • 800-369-4768 • Fax: 719-548-1902
WWW: http://www.dmwworldwide.com

## HostCHECK for UNIX: New Security for E-Commerce
## White Paper
Dr. Bruce V. Hartley
DMW Worldwide

**HostCHECK™**

**E-Business Creates Security Risks**

You can reach millions of potential customers through e-business. At the same time, you create new entry points into your systems, potential gateways for security breaches and risks. In 1997, DMW Worldwide conducted over a dozen controlled penetrations of major industries. In every case, we were successful—and in each case, the problem hinged on inadequate host security.

Internet Firewalls protect perimeters. In the cases noted above, the firewall was inadequate protection. Clearly, better host-based security is needed to contain security problems if your firewall is breached, and to identify intrusions if they do occur.

**The Enterprise Security Model: A Layered Approach**

Many organizations have been lulled into a false sense of confidence with their computer security configuration by using firewall products. While properly configured firewalls prevent people from breaking into the computer from the outside, perimeter security is only one component of the Enterprise Security Model, which is based on a layered approach to security. This layered approach combines perimeter security, host-based security, security education awareness, and an incident response capability, which are interconnected by an Enterprise Security Policy. If the security inside the company is weak, it still remains an easy target.

In light of the aging of available security tools, and the internal weaknesses that are generated as a result, a more advanced tool is needed to address UNIX security. DMW Worldwide is filling the void with HostCHECK™.

**What is HostCHECK?**

HostCHECK is a UNIX computer security program developed in ANSI C. It is arranged in a modular/integrated form and consists of nine computer security modules and several utilities, each testing a different aspect of the computer's security. These modules are tightly integrated to form a complete security package.

The program was designed to focus on internal computer security, that is, locating the security problems that can be exploited from a user already logged into the computer system. HostCHECK is able to keep intruders out of the computer system by identifying improper configuration and making the problem easy for the system administrator to repair.

**HostCHECK is an Integrated Security Package**

Today, many mission-critical business functions are run on UNIX servers. HostCHECK integrates features normally offered as stand-alone security products and incorporates them into one package with up-to-date security and vulnerability information. HostCHECK addresses many of the deficiencies found in other UNIX security tools, providing an easy-to-use interface and user-friendly output reports. This enables less-experienced UNIX administrators to use and benefit from the tool.

| Service | Title | Protocol | Port | Status |
|---|---|---|---|---|
| walid/1 | Unknown Ser... | rpc/datagra... | UNKWH | ACTIVE |
| uucp | Unix to Unix ... | tcp | 117 | ACTIVE |
| tftp | Trivial File Tr... | udp | 69 | inactive |
| telnet | Telnet Shell ... | tcp | 23 | ACTIVE |
| talk | User to User ... | udp | 517 | ACTIVE |
| systat | System Stats | tcp | 11 | inactive |
| sprayd/1 | Unknown Ser... | rpc/datagra... | UNKWH | ACTIVE |
| shell | Remote Shell... | tcp | 514 | ACTIVE |
| rusers d/2-3 | Unknown Ser... | rpc/datagra... | UNKWH | inactive |
| rstatd/2-4 | Unknown Ser... | rpc/datagra... | UNKWH | ACTIVE |
| rquotad/1 | Unknown Ser... | rpc/datagra... | UNKWH | ACTIVE |
| rexd/1 | Unknown Ser... | rpc/tcp | UNKWH | inactive |
| netstat | Network Stat... | tcp | 15 | inactive |
| name | Name Server ... | udp | 42 | ACTIVE |
| login | Remote Login... | tcp | 513 | ACTIVE |
| ftp | File Transfer ... | tcp | 21 | ACTIVE |
| finger | Finger Indent... | tcp | 79 | ACTIVE |
| exec | Remote Exec... | tcp | 512 | ACTIVE |
| echo | Internal Echo... | tcp | 7 | ACTIVE |
| echo | Internal Echo... | udp | 7 | ACTIVE |
| discard | Internal Disc... | tcp | 9 | ACTIVE |
| discard | Internal Disc... | udp | 9 | ACTIVE |
| daytime | System Time... | tcp | 13 | ACTIVE |

**HostCHECK Screen with Security Tab Pull-down**

48

**HostCHECK Provides Immediate Security Improvement**

HostCHECK is set apart from other UNIX security tools by its AutoCorrect capability, which can immediately improve the security posture of a UNIX host. Through interactive dialogue boxes, HostCHECK describes the vulnerabilities found on the system, suggests corrective measures, and provides the rationale for the corrections. The system administrator may then request that HostCHECK implement the corrective measures to eliminate the identified vulnerability.
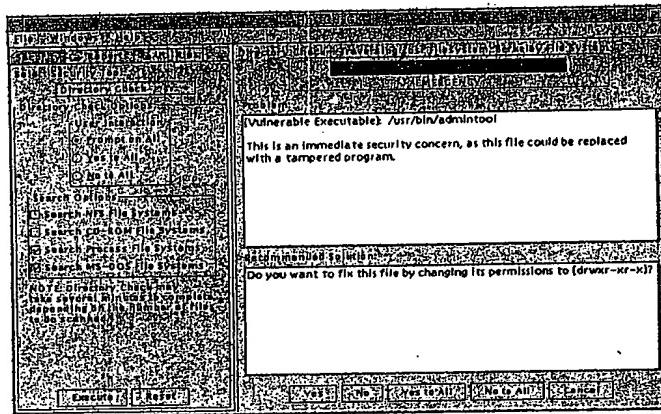


**HostCHECK AutoCorrect Screen**

49

password scheme the system uses, file-system construction, etc. HostCHECK uses this information during the initial set-up of the software. The result is a security baseline customized to the current system configuration.

**A Friendly GUI**

HostCHECK's JAVA-based GUI makes it easy to use for all system administrators. Interactive dialogue boxes explain the vulnerabilities found, give a rationale for correction, and prompt the user to make the corrections immediately. Color-coding of screens help alert users when problems do arise. HostCHECK may be executed three different ways: through the GUI buttons and pull-down menus, through Text Mode Interface, or through the use of UNIX commands.

**Usable Reports**

Most security tools are hard for the novice administrator to use, and their output is both cryptic and difficult to understand. Many of these tools identify problems, but leave determining the solution to the system administrator. To further compound the problem, inexperienced administrators may tend to ignore suggested corrections due to time constraints or lack of experience.

HostCHECK's output reports are easy to read, descriptive of the location of the problem, the dangers involved, and are useful in educating the system administrator on security practices. Other reporting systems are hard to understand, and do not explain the problem in detail. HostCHECK's ReportIT utility generates reports in non-technical, plain English that explain the nature and severity of the problem found, what can be done to fix the problem, and where patches and additional information about the problem can be found on the Internet. Comprehensive or specific reports can be specified, and these reports can be printed, stored as text files, or mailed to a specified user account. All reports are archived and comparisons can be made between new and previous reports, aiding in intruder detections.



### CrackIT Password Report
Copyright (C) 1998 by DMW Worldwide, LLC

Password Security Table

| User Account | User Name | Password |
|---|---|---|
| mclark | Mclark | mclark |
| test | Test | test |
| rsasoka | Rsasoka | rsasoka |
| victim | Victim | secret |
| jraffold | Jraffold | Karen |
| cmarros | Cmarko | cmarko |
| rwalker | Rwalker | tennis |
| dlowrey | Dlowrey | train |
| rpierce | Rpierce | secret |
| arendon | Arendon | NO PASSWORD |
| gleiman | Gleiman | NO PASSWORD |
| gltodd | Gltodd | NO PASSWORD |

**HostCHECK Password Report Screen**

**Nine Security Modules**

HostCHECK consists of nine security modules along with supporting utilities. Each security module focuses on a different aspect of computer security, and has a pathway of communication with each of the other

modules. The security modules are:

- Directory Check
- Integrity Check
- User Manager
- User Check
- CrackIT
- RemoveIT
- ReviewIT
- Network Check
- Configuration Check

HostCHECK was designed to be a comprehensive host security system, rather than providing a single solution to isolated security problems. Thus, HostCHECK contains more functions than other security programs currently on the market.

The graph below compares the number of security exams run by
HostCHECK to other comparable products.



**Number of Security Exams**
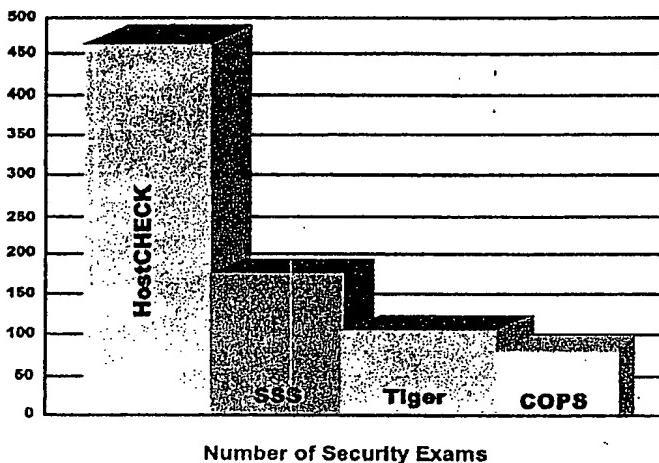
**Directory Check
Locates Security
Flaws and
Prompts
AutoCorrection**

Directory Check is the true heart of HostCHECK, and is one of the most
unique and effective features of the package. The Directory Check searches
through all the files in a UNIX computer's file system to locate security
flaws developed from accidents, improper configuration, pre-existing bugs,
or even hacker intrusions.

The Directory Check runs a series of tests to determine if the information in
"critical" files has been tampered with. A "critical" file is any file that alters
or changes the system's security (such as a set user id or set group id file, a
device driver, or a configuration file flagged as critical by Configuration
Check). Each critical, security file's checksum, SHA-1 message digest,
length, permissions, ownership, and group are compared against the
Directory Check's previous run[1]. Other files are examined for secure file
permissions to ensure new files of security concern are tracked. Likewise,
device drivers are examined for correct association of file rights and
ownership to the major and minor number of the device.

---

[1] On initial installation, HostCHECK compares against a database derived from the original manufacturer's
distribution media.

53

**HostCHECK Directory Check Screen**

### AutoCorrect Allows Immediate Correction of Security Problems
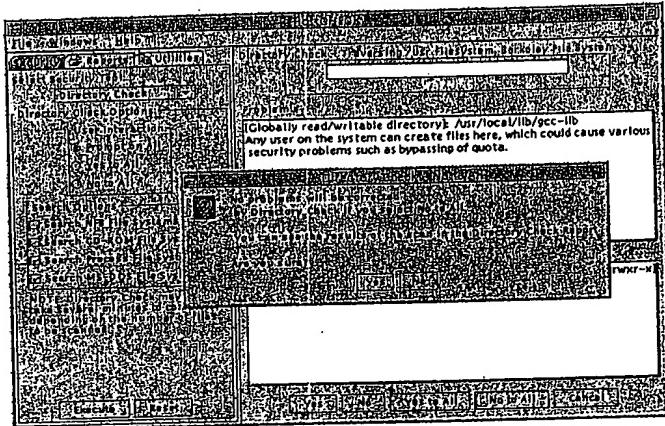
One of the more useful features of the Directory Check is its ability to explain a found error in detail and describe how the problem impacts the overall security of the system. In most cases, it also prompts the system administrator to implement the suggested corrective measure for the security problem. This AutoCorrect feature ensures that quick solutions for the system administrator are available as soon as the problem is identified. Another outstanding feature of the Directory Check is its speed. Written in ANSI C using efficient programming techniques, it usually takes only a minute or two to scan file systems consisting of tens of thousands of files.[2]

---

[2] In a benchmark test, a files system containing fifty thousand files on a 80486-50DX computer took only 84 seconds.

The graph below illustrates the speed of HostCHECK compared to other products.



**Security "Consultant Grade" Exam***

* Tested on Sparc station LX with 32 MB of RAM
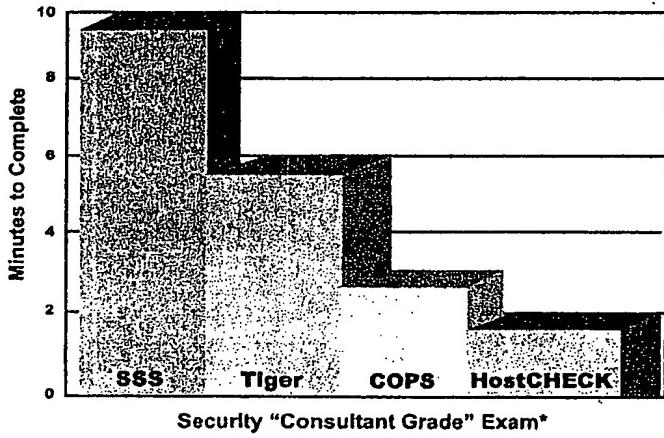
**Integrity Check
Uses Extensive
DMW Vulnerability
Database**

Many UNIX security vulnerabilities are included in the initial operating system and were discovered after public usage. Even if an administrator was to patch these security vulnerabilities, reinstalling the operating system or missing a critical security announcement may allow the system security to be compromised. The Integrity Check module identifies and keeps track of these vulnerabilities.

The Integrity Check module searches for security problems that are associated with a specific platform and notifies the system administrator that a problem exists. Information is then made available to the system administrator as to the nature of the vulnerability and where the appropriate patch may be found.

An extensive database of existing security vulnerabilities and patches is an integral part of HostCHECK and is used to run a comparison against the computer's integrity. This information comes from the DMW Vulnerability Database, which consists of information pooled from many sources, such as advisories, trade journals, hacker publications, Computer Emergency Response Team (CERT®) reports, and in-house vulnerability testing and analysis. Currently, the DMW Vulnerability Database is one of the largest on the market, containing close to 2000 vulnerabilities.

The graph below shows the HostCHECK's Vulnerability Database compared to those of other products.



**Number of Vulnerabilities Checked**
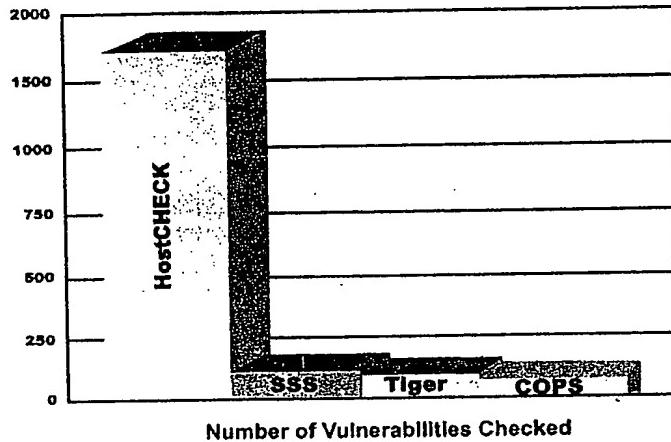
**User Manager Allows Interface to Run Across Multiple UNIX Platforms**

Good user management systems are difficult to find and typically are not portable. To address this need, the User Manager module was created to provide an easy-to-understand user management system that not only identifies security problems associated with user accounts, but also allows user accounts to be created in proper grounds, making user management easier and more secure.

The User Manager Tool provides extensions to the basic concept of groups in UNIX, which allows several groups of users, each with different styles of home directories, access privileges, and shells.

**User Check Monitors User Access**

The User Check performs routine security checks to determine whether or not system users have allowed easy access into the machine from the outside (by way of their ".rhost" file), or if a security problem exists, such as users not owning the rights to their home directory or files within their home directory.

**CrackIT Ensures Password Security**

Many times the security of a system is compromised by poor passwords. Easy-to-guess passwords such as "system," "password," or "secret" can allow a breach in security. Even if a hacker were to steal the system's password file and attempt to crack passwords, it is critical to have a password protection system capable of attaining very high speeds and providing a reasonably high certainty that the system's users have difficult-to-guess passwords.

HostCHECK's CrackIT module ensures the security of your system's passwords by examining and identifying all existing weak passwords. Once these have been changed and rechecked for security, the probability of a security breach due to poor passwords is greatly reduced.

| | |
|---|---|
| CASE STUDY: Passwords Cracked by HostCHECK Scrutiny | During a recent penetration engagement, 800 standard UNIX DES passwords were audited. Two computers split the task: an UltraSparc 1 and a Pentium 166. Using the DMW CrackIT software package, the audit performed over one and a half billion password cracking attempts in slightly over three days. The audit resulted in passwords for 35% of the accounts, and the first password was obtained less than three minutes into the audit. In a separate audit performed on a Pentium 90 notebook over the course of a weekend, 264 UNIX accounts were examined, yielding 212 guessed passwords (80%). |

**Maximum CPU Time Utilized**

The CrackIT module is optimized to utilize maximum CPU time available to achieve the highest possible speeds. The password checking algorithm incorporates such time saving techniques as "same salt" and "similar salt" cryptographic shortcuts.

Besides guessing information from dictionaries, the words may also be "filtered" through various change techniques to create derivative words that might not be found in a dictionary but are still easy to guess. For example, the letter "t" looks like the symbol "+". Therefore, if the filter were to receive the word "testing," it would generate and test the word "+es+ing." Currently, there are 8,192 combinations in the filter and others still planned.

Also, the GECOS password field (the segment of the password line usually containing the user's name and office telephone number) is also used to generate password guesses. A total of 363 combinations are searched, and three common generations from a line of information such as "John Q. Public, 555-3090" would be "John3090", "PublJohn", and "Publ5553."

CrackIT is also available separately or packaged with the RemoveIT and ReviewIT tools.

**RemoveIT Ensures File Security**

Sensitive information often must be removed from the system. The administrator must ensure that the data are destroyed; merely deleting the file does not ensure the data are inaccessible by other means.

57

The RemoveIT utility is a secure means for deleting sensitive information and conforms to DoD specifications. The file containing the sensitive information is removed by overwriting the file multiple times, each time verifying that the information has been properly deleted. With this method, users of the system cannot recover the file.

RemoveIT is also available separately, or packaged with the CrackIT and ReviewIT tools.
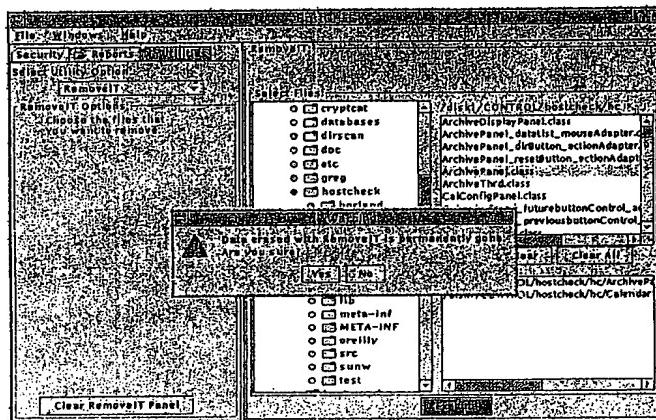


**HostCHECK RemoveIT Screen**

**ReviewIT Provides More Intrusion Detection Capability**

The ReviewIT tool allows the system administrator to review audit trail settings, and to interactively change audit trail settings. ReviewIT allows the administrator to evaluate current audit trails against recommended audits, configure audit trails on the fly, and further define audit requirements, audit specific users, and create usable, intelligible audit reports. ReviewIT is also available separately, or packaged with the CrackIT and RemoveIT tools.

**Security Check Tracks and Corrects File Permissions**

A typical UNIX operating system is comprised of over 10,000 files. It is virtually impossible for a system administrator to know all the suggested permissions for each file. The Security Check compares the permissions and ownership of system files against a database of DMW's recommended permissions for those files.

If an intruder were able to modify just one of the files, it is possible for he or she to create an easy access path into the operating system. Also, normal daily activity could inadvertently change the permissions of one of these files in such a way as to allow a casual observer to gain additional access. Security Check is a way of checking and preventing this.

58

**Network Check Monitors Intrusions**

The Network Check is designed to examine and display aspects of the computer's network security configuration. If the Network Check determines the computer is running in "promiscuous" mode (a state referring to the Ethernet adapter), it means a "sniffer" program is running on the computer system. In this situation, it is likely that an active intrusion is taking place on the network.

During Network Check, HostCHECK automatically executes the Port Scanner to determine TCP services are running on the host. A TCP service is any sessions-oriented communication where the computer system will allow outside computers to initiate communication with it (i.e., FTP, Telnet, and Remote Login Services). The Internet server configuration file is summarized and displayed to show all network services being provided.

**ScheduleIT Aids in Intrusion Detection and Monitoring**

Business environments as well as operating systems are dynamic. With HostCHECK, companies have a powerful way to not only assess and correct security problems, but to monitor and protect their security posture over time. HostCHECK's ScheduleIT and ReportIT utilities allow users to run checks on a regular basis and to compare new reports to archived reports. Using ReviewIT and Network Check to search for "sniffers" running on the host, check for insecure entries in the network configuration, and to review and modify audit trail settings provides businesses with the strong intrusion detection and control they need.
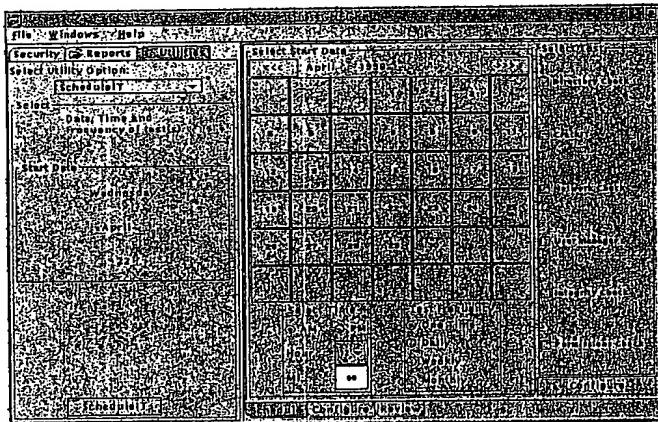


**HostCHECK ScheduleIT Screen**

**HostCHECK Fills
the Need for
Better Security**

CERT® has reported thousands of computer incidents annually affecting over 146,000 computers.[3] In fact, 520 U.S. companies reported a total loss of $136 million from computer crime and security breaches in 1997, up 36% from the previous year.[4] Because of the vast numbers of UNIX platforms that exist controlling the flow of information on the Internet, the need for quality computer security products has become clear.

HostCHECK was designed to fill this need. It's modular design makes it capable of including any security programs needed for the ever-expanding growth of UNIX security issues. HostCHECK is focused on portability, diversity, and user-friendliness, with modules that are more powerful and efficient than existing security tools. The key differences between HostCHECK and other UNIX security tools is HostCHECK's ability to report potential security vulnerabilities, and its AutoCorrect mechanism, enabling the system administrator to take the appropriate corrective actions. Thus, HostCHECK produces immediate improvements in the overall security posture.

**Easy and
Frequent Updates**

HostCHECK may also be easily updated. Because HostCHECK relies on its Vulnerability Database for the majority of its security checks, updates are easily installed since they do not require rewriting the code. HostCHECK's Vulnerability Database is updated frequently, allowing HostCHECK to remain a cutting edge product and to continue to provide customers with the powerful internal security they need to conduct e-business.

**HostCHECK: The
Power to Protect**

If you use UNIX, you need HostCHECK. HostCHECK complements existing security mechanisms, such as firewalls, to create a secure environment that enables e-business. Why take the risk? Why be a statistic? Get HostCHECK, and get the power to protect.

---

[3] From the Computer Emergency Response Team 1997 Annual Report, the CERT® Coordination Center handled 2,134 computer security incidents affecting over 146,000 sites during 1997.
[4] Study published by the Computer Security Institute and the FBI, reported in InternetWeek, March 28, 1998.

# HostCHE〔 powers your comp〔 〕y for e-business

## The Power to Protect

HostCHECK's security modules are complemented by powerful features: intelligent reporting, a friendly GUI, and the ability to make immediate security improvements. Developed in ANSI C for both speed and portability, HostCHECK's modular architecture gives it optimal performance—HostCHECK completes its "security consultant grade" examination in less than two minutes compared to ten for its nearest competitor.

## User-Friendly Power

HostCHECK's JAVA-based GUI makes it easy to use even for novice administrators. Interactive dialogue boxes explain the vulnerabilities found, give a rationale for the correction, and prompt the user to make the corrections immediately. Color-coding of screens helps alert users when problems do arise. HostCHECK may be executed in three different ways: through the GUI buttons and pulldown menus; through Text Mode Interface; or through the use of UNIX commands.

Designate additional security-critical files with MarkIT. Make immediate security improvements with AutoCorrect. Run regular and automated checks with ScheduleIT.
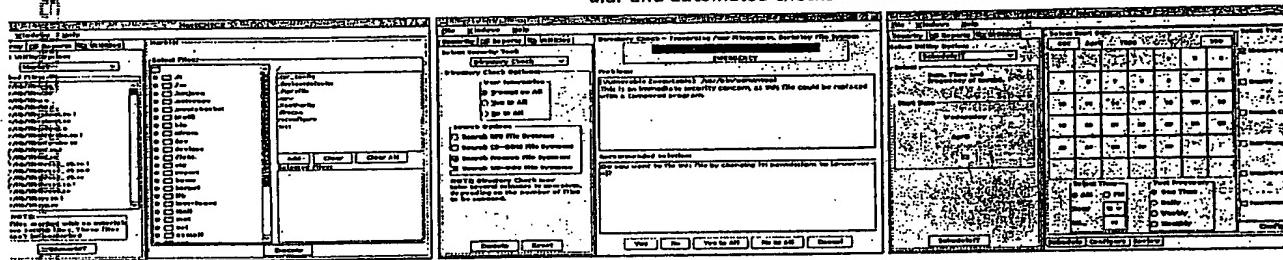


## Intelligent Reporting

Most security tools have output that is highly cryptic and hard to understand. HostCHECK's ReportIT utility generates reports in non-technical, plain English and explains the nature and severity of the problem found, what can be done to fix it, and where patches and additional information about the problem are located. Comprehensive or specific reports can be specified, and these reports can be printed, stored as text files, or mailed to a specific user account. All reports are archived and comparisons can be made between new and previous reports, adding critical intrusion detection capability to the security tool set.

## Immediate Security Improvements

HostCHECK's extensive database of close to 2000 vulnerabilities and its AutoCorrect feature give you the power to secure your system immediately. The Vulnerability Database is compiled from both public and semi-public sources and each vulnerability is cross-referenced to advisories, patches, exploits, and functions. AutoCorrect is designed so that when HostCHECK detects a problem, it identifies the problem on screen for the user, recommends a solution to improve security, and then provides the rationale for that correction. This gives the System Administrator the ability to make immediate improvements to the company's security posture.

# ...to protect

HostCHECK powers your company for e-business with a suite of nine security tools integrated into one easy-to-use program.

**Directory Check:** Searches every file on the system for security problems. Detects unsecure file permissions as well as file tampering, newly created "setuid" files, Trojan horse installations, and viruses. Checks for unusual device permissions, ownership, and groups. Fingerprints all security-critical files using checksums, file length, and SHA-1 secure hash. Directory Check is very fast, checking the security of up to 30,000 files in 60 seconds. Directory Check includes the Permissions Check utility, which checks defaults on 1400 possible mis-configured files. Permissions Check allows automatic changes and has a rollback capability, allowing the administrator to try many security approaches safely. AutoCorrect capability prompts users to correct identified problems and provides an explanation of each vulnerability discovered.

**Integrity Check:** Checks for pre-existing security problems by cross-referencing against the extensive DMW Vulnerability Database. As part of HostCHECK's AutoCorrect capability, the user is prompted to immediately correct any identified problems. For those vulnerabilities that HostCHECK can not immediately correct, HostCHECK provides the WWW page references where more information and the vendor patch are located. To date, CERT has issued 827 vulnerability reports. DMW's Vulnerability Database contains close to 2,000 vulnerabilities.

**Configuration Check:** Detects system characteristics, enabling HostCHECK to configure itself correctly; HostCHECK's ability to detect subtle details in the configuration of the computer makes it very portable across platforms. Identifies files considered critical to the security configuration of the computer or files that, if they are tampered with, can be used for easy penetration into the system. Automatically identifies the properties of the system for use in the initial software set-up.

**Network Check:** Searches for sniffers running on the host and for unsecure entries in the network configuration. Displays all services running on the host, including those not registered with the Internet daemon.

**User Manager:** Many security incidents are due to poor system administration. HostCHECK's User Manager allows system administrators to create interfaces that can be run across multiple UNIX platforms and to manage account and group structures, eliminating the creation of new security vulnerabilities when new accounts are created.

**CrackIT:** Possibly the most effective password-cracking tool ever developed. CrackIT is extremely flexible; capable of cracking passwords locally or remotely, it is able to communicate with a wide range of different services. Options include selecting from multiple dictionaries containing

millions of words in eighteen languages. CrackIT has advanced filtering and GECOS ability.

**ReviewIT:** Allows administrators to both review and interactively modify audit trail settings. Allows evaluation of current audit trails against recommended audits, configuration of audit trails on the fly, further definition of audit requirements, auditing of specific users, and creation of usable, intelligible audit reports.

**User Check:** Checks insecurities associated with users' accounts and home directories; detects open home directories, Trojan horses, and other security risks.

**RemoveIT:** Obliterates a file from online media by overwriting its location repetitively. Follows the Department of Defense "Blue Book" removal guidelines for classified data remnants as set by the National Computer Security Center. Overwrites files using specific bit patterns and simple text strings.

### Customized Protection

You can extend the flexibility of HostCHECK's protection with the MarkIT utility, allowing the administrator to select security-critical files, web pages, etc., for a security scan by HostCHECK. The unmark utility can remove protected files from the checking list, but contains a fail safe that prevents the exclusion of any security-critical components.

### Constant Monitoring and Detection

Your business environment is dynamic, as is your operating system. HostCHECK provides a powerful mechanism to assess and correct security problems, and to monitor and protect your security posture over time. HostCHECK's ScheduleIT and ReportIT utilities allow you to run checks on a regular basis and to compare new reports to archived reports. Network Check searches for sniffers running on the host and checks for insecure entries in the network configuration. ReviewIT reviews and modifies audit trail settings to provide your business with the intrusion detection and control you need in today's changing environment.

HostCHECK for UNIX is an integrated collection of security programs combined with DMW's state-of-the-art, extensive Vulnerability Database, a friendly and comprehensive reporting mechanism, and an intuitive and

easy-to-use graphical user interface. HostCHECK goes far beyond just reporting security problems—HostCHECK's AutoCorrect mechanism allows immediate security improvements to take place within the tool, achieving an improved UNIX security posture the moment a security problem is identified. HostCHECK for UNIX was developed in ANSI C to optimize speed and portability, requires 32 MB of RAM, and now supports Solaris 2.3 or higher, SunOS 4.1.X, FreeBSD, HP-UX v9 and v10, Digital UNIX 2.X, Irix 6.X, AIX 4.X, and Linux 2.X.

### DMW Worldwide

DMW Worldwide develops electronic business solutions that adapt quickly to changing technology and customer demands. Timarou™, the company's family of integrated solutions, provides proactive customer care, adaptive network care, and convergent billing. By combining these products with strategic business and network services, clients receive a seamless suite of solutions to help them efficiently and securely conduct business over the network, and to manage customer needs proactively. Timarou is flexible to adapt to dynamic business environments and redefine what is possible with

e-business—exceptional customer relationships, real-time billing, end-to-end information services management, and secure world-wide communications.

A secure infrastrature means business. Hostcheck powers your company for e-business.

# HostCHECK Security Process

**HostCHECK**

HostCHECK™



Diagram labels: Directory Check, Integrity Check, CrackIT, Network Check, UserCheck, AutoCorrect, MarkIT, ReviewIT, RemoveIT, User Manager, CONFIGURATION CHECK, ASSESS, EVALUATE, DMW's Vulnerability Database, CORRECT/IMPROVE SECURE POSTURE, MANAGE CONFIGURATION, PROTECT & MONITOR (IDS), ReviewIT, ScheduleIT, MarkIT, Network Check, User Manager

- Assess current security of system.

- Immediately correct problems to create improved security posture.

- Protect system against intrusion to maintain improved security.

**DMW WORLDWIDE**

5
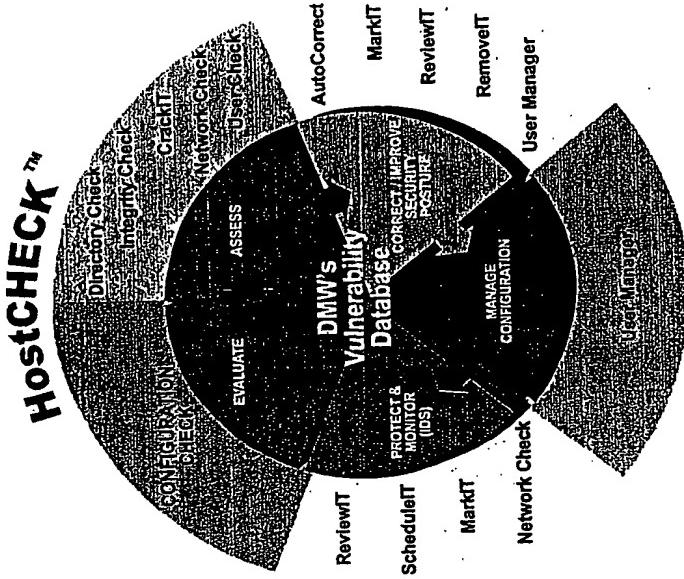
# Directory Check

- Searches EVERY file on the file system for security problems.

- Detects unsecure file permissions.

- Detects file tampering, newly created "setuid" files, Trojan horse installations, and viruses.

- Detects unusual device permissions, ownership, and groups.

- Includes Permissions Check utility.

- Allows for AutoCorrect capability.

**DMW Proprietary**

8

# Directory Check Screen

HostCHECK

DMW WORLDWIDE

HostCHECK

File  Windows  Help

Security  Reports  Utilities

Select Security Tool:

Directory Check

Directory Check Options:

User Interaction
- ⦿ Prompt on All
- ○ Yes to All
- ○ No to All

Search Options
- ☐ Search NFS File Systems
- ☐ Search CD-ROM File Sys
- ☑ Search Process File Syst
- ☑ Search MS-DOS File Syst

NOTE: Directory Check may take several minutes to com depending on the number of files to be scanned.

Directory Check – Traversing /usr Filesystem; Berkeley File System

Problem:

[Globally read/writable directory]: /usr/local/lib/gcc-lib
Any user on the system can create files here, which could cause various
security problems such as bypassing of quota.

Directory Check – No to All

No problems will be corrected
by Directory Check if you select No to All.

You can see the results of the scan in the Directory Check Report.

Are you sure?

Yes    No

rwxr-x

Yes    No    Yes to All    No to All    Cancel

Execute    Reset

DMW Proprietary

9

# AutoCorrect

- Allows for immediate security improvement.

- Displays screen prompt when HostCHECK detects a vulnerability.

- Identifies vulnerability, suggests correction, and provides rationale for correction.

- Prompts System Administrator to make correction.

**DMW**
DMW WORLDWIDE

DMW Proprietary

11

# CrackIT

- Uses the most optimized password cracking algorithms known to the public.

- Extremely flexible.

- Capable of cracking passwords locally or remotely and communicates with a range of different services.

- Huge dictionary containing millions of words spanning 18 languages.

- Advanced filtering and GECOs ability.

18

83

# RemoveIT

- Obliterates a file from online media.

- Follows DoD "Blue Book" removal guidelines set by the National Computer Security Center.

- Overwrites files using specific bit patterns and simple text strings.

21

# RemoveIT Screen

HostCHECK

DMW WORLDWIDE

**HostCHECK**

File  Windows  Help

Security | Reports | Utilities

Select Utility Option:

RemoveIT ▼

RemoveIT Options:
Choose the files that
you want to remove.

RemoveIT:

Select Files:

/disk1/CONTROL/hostcheck/hc

- cryptcat
- databases
- dirscan
- doc
- etc
- greg
- ● hostcheck
- borland

ArchiveDisplayPanel.class
ArchivePanel_datalist_mouseAdapter.c
ArchivePanel_dirButton_actionAdapter
ArchivePanel_resetButton_actionAdapt
ArchivePanel.class
ArchiveThrd.class
CalConfigPanel.class
futurebuttonControl_a
previousbuttonControl
class

Clear All

lear

**Warning**

Data erased with RemoveIT is permanently gone.
Are you sure?

Yes   No

- lib
- meta-inf
- META-INF
- oreilly
- src
- sunw
- test

0U/hostcheck/hc/ArchiveP
0U/hostcheck/hc/Calendar

Execute

Clear RemoveIT Panel

DMW Proprietary

22

87

# ReportIT

- Most security tools have cryptic output.

- HostCHECK has intelligent reporting.

- HTML-based, non-technical, plain English reports.

- Explains nature and severity of problem found.

- Identifies location of patches and additional information on WWW.

- Prints reports, stores as text files, mails to specific user account.

DMW Proprietary

24

(Submitted in Appendix A of...

# MarkIT

- Extends the protection of HostCHECK.

- Designate security-critical files, web pages for security scanning.

- Unmark removes protected files from checking list.

- Fail-safe prevents exclusion of security-critical components identified by HostCHECK.

# MarkIT Screen

**HOSTCHECK**

File  Windows  Help

Security  Reports  Utilities

MarkIT

Select Utility Option:

MarkIT

Marked Files:

/usr/lib/libw.so.1
/usr/lib/libw.so
/usr/lib/libw.a
/usr/lib/libvt0.a
/usr/lib/libvolmgt.so.1
/usr/lib/libvolmgt.so
/usr/lib/libvolmgt.a
/usr/lib/libtnfprobe.so.1
/usr/lib/libtnfprobe.so
/usr/lib/libtnf.so.1
/usr/lib/libtnf.so
/usr/lib/libthread_db.so.1
/usr/lib/libthread_db.so.0
/usr/lib/libthread_db.so
/usr/lib/libthread.so.1
/usr/lib/libthread.so
/usr/lib/libsys.so.1
/usr/lib/libsys.so

NOTE:
Files marked with an asterisk
are SetUID files. These files
can't be unmarked

Unmark

Select Files:

○ .dt
○ .fm
○ .hotjava
○ .netscape
○ .wastebasket
○ (null)
○ bin
○ cdrom
○ dev
○ devices
⊕ disk1
● CONTROL
  ○ archive
  ○ bin
  ○ certify
  ○ composite
  ○ config
  ○ cryptcat
  ○ databases
  ○ dirscan
  ○ doc
  ○ etc

/disk1/CONTROL/hostcheck/lic

ArchiveDisplayPanel.class
ArchivePanel_datalist_mouseAdapter.class
ArchivePanel_dirButton_actionAdapter.cla
ArchivePanel_resetButton_actionAdapter.
ArchivePanel.class
ArchiveThrd.class
CalConfigPanel.class
CalendarPanel_futurebuttonControl_actio
CalendarPanel_previousbuttonControl_ac
CalendarPanel.class

Add  Clear  Clear All

Selected Files:

/disk1/CONTROL/hostcheck/hc/ArchivePand
/disk1/CONTROL/hostcheck/hc/CalendarPa
/disk1/CONTROL/hostcheck/hc/ArchiveDisp
/disk1/CONTROL/hostcheck/hc/ArchivePand

Execute

DMW Proprietary

DMW WORLDWIDE

28

# ScheduleIT

HostCHECK

- Monitor and correct security posture over time.

- Run checks on a regular, pre-set basis.

- Email reports to specified user addresses.

- Run any report, any time, automatically.

DMW
DMW WORLDWIDE

DMW Proprietary

29

# ScheduleIT Screen



HostCHECK

File   Windows   Help

Security | Reports | Utilities

Select Utility Option:
ScheduleIT

Select
Date, Time, and
Frequency of test(s).

Start Date
Wednesday
April
22

ScheduleIT

Select Start Date
<<<   April   1998   >>>

| 1 | 2 | 3 | 4 |
| 8 | 9 | 10 | 11 |
| 15 | 16 | 17 | 18 |
| 22 | 23 | 24 | 25 |
| 29 | 30 | | |

5 | 6 | 7
12 | 13 | 14
19 | 20 | 21
26 | 27 | 28

Select Test
☑ Directory Check
☑ CrackIT
☑ Network Check
☑ User Manager
☑ Integrity Check
☐ Permissions Check

Configure

Select Time
◉ AM   ○ PM
Hour  12 ▼
MI...  00

Test Frequency
◉ One Time
○ Daily
○ Weekly
○ Monthly

Schedule | Configure | Review

DMW Proprietary

30

# DMW Vulnerability Database

- Made from public and semi-public vulnerability sources.

- Composed of multiple databases.

- Contains nearly 2,000 vulnerabilities.

- References vulnerabilities to advisories, patches, exploits, and function.

Number of Vulnerabilities Checked

HostCHECK

SSS

Tiger

COPS

2000 | 1500 | 1000 | 750 | 500 | 250 | 0

DMW Proprietary

32
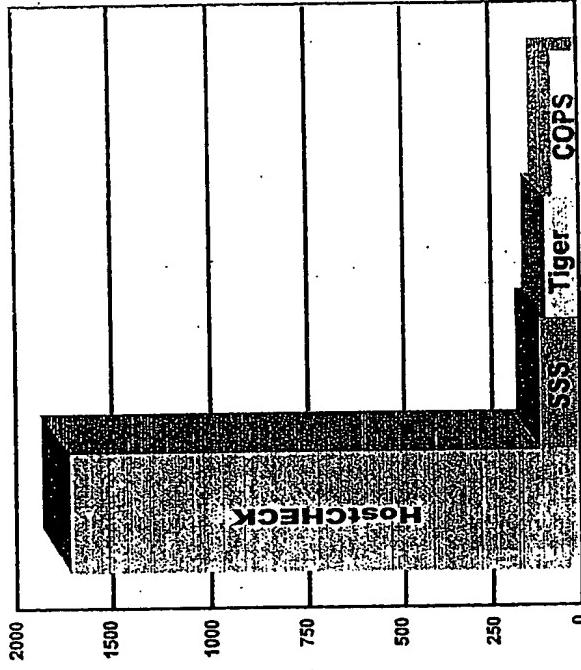
Phase 10:      Brute force, all possible guesses shall be attempted (within the scope limitations of the encryption.)

**Unix Password Checking Limitations**

Not all methods of password examination are needed for the HostCHECK software package, so the following methods shall be needed within the UniCrack software package:

- DES (Digital Encryption Standard) based password cracking

**Unix Password Storing Awareness**

Password files are stored differently on different UNIX flavors and configuration management methods. The following methods shall be implemented in order to handle the majority of the UNIX platforms:

- Standard /etc/passwd file standard
- /etc/master.passwd file standard
- /etc/shadow file standard

## User Management Requirements

**User Security Examinations**

Security tests shall be performed in order to determine if a user's security is poor. The tests performed by the user security system shall be:

- User owns his/her home directory
- User's group is associated with his/her home directory
- User's home directory no longer exists
- User doesn't have a possibly insecure $(HOME)/.rhosts
- User doesn't have a possibly insecure $(HOME)/.netrc
- User has a unique user ID number
- User owns his/her mailbox
- User's mailbox allows only user to read his/her mail.
- System administrator can easily locate files on system associated with single user.

**User Management Functions**

Management functions will be provided in order to allow management of users to be easier. These functions shall be:

- Users can be easily created .
- Users can be created in a simple "template" style.
- Users can be easily deleted

## File Destruction

An ability for files to be removed from the system following the Department of Defense data remnant removal procedures shall be implemented.

## Network Examinations

In order to allow examinations of the host's network configuration for security concerns, the following network security features shall be performed:

- TCP wrapping shall be used to provide a viable audit trail for incoming network connections.
- Promiscuous mode shall be checked for in order to determine if a sniffer is running on the host.
- Examination of a "+" in the computer's host.equiv to prove it is not present.

## Vulnerability Testing

Vulnerability testing involves looking for pre-existing security weaknesses that may be inherent with the operating system or the way the operating system or application software is configured by default.

### Vulnerability Test Methods

The following tests shall be used to "narrow down" the search to determine if a vulnerability could exist on the computer:

- The file is older than the patch.
- The file has permissions that may be associated with an insecure file.
- The file "passes" a vulnerability check by observing the program's behavior.
- The file is owned by a specific user associated with the vulnerability.
- The file is owned by a specific group associated with the vulnerability.

### Vulnerability Database

The database used for the Vulnerability testing shall be from the DMW Vulnerability Database (approximately 1,300 possible security vulnerabilities)

## Intrusion Detection/Backdoor Examination

In order to determine if file tampering has taken place, the following intrusion detection capabilities shall be performed in order to determine if files have been created which add to suspicion that the user is attempting to hiding files or otherwise abusing system resources:

- Suspicious usage of ".." or "..." directories
- Suspicious usage of .plan or .fingerrc
- Suspicious usage of /usr/spool/uucppublic
- Suspicious creation of a new setuid file
- Suspicious creation of a new setgid file

## Common Misconfiguration

In order to prevent whether a security problem may exist as a result of improper configuration, the following tests shall be performed:

- Invalid use of # to comment out password entries
- Exporting file-systems to everyone
- Insecure PATH setting for root access

# System Architecture

## System Diagram

### Highest Level Component Interaction

HostCheck consists of eight security modules and five utility programs which interact in order to identify security problems, report the problems, and take corrective action if necessary. This software package could be thought of as a collection of different yet important tools.

The Eight Security programs are the Directory Scanner, Integrity Checker, Password Cracker, Configuration Detection System, System Profiler, User Manager, Network Manager, and Wipe. Each of these utilities provides a different aspect of security to the package.

110
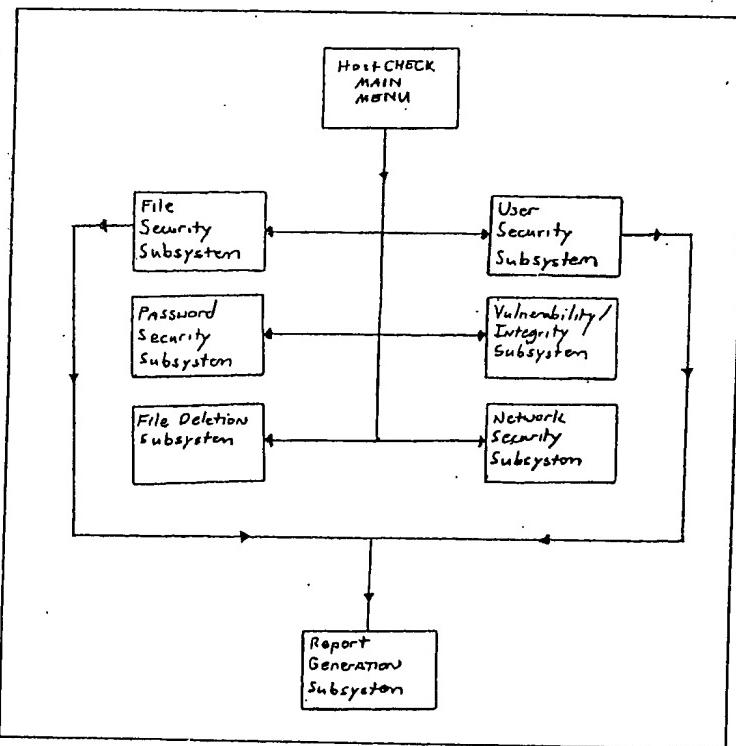
## System Software Components

### Configuration Detection System

The Configuration Detection System identifies files which are "critical" (i.e., files which are necessary to maintain the system's security) and locates unusual features of this particular computer. The files which are identified are transferred to the File Security System to aid in securing the system properly. This program needs to be run only once per installation.

Because the configuration of each computer system is different, and the system administrator doesn't want to waste considerable time explaining to the security software program all of the differences between the current computer's security and that of a default machine, the Configuration program was also designed to accommodate other security issues besides critical files.

The configuration security component is merely an interpreter that calls a set of script programs. Each of the scripts focuses on a single area of configuration concern: executables, internet connectivity, password management, X Windows management, and library management to name a few possible areas of concern.

There is a second purpose that this program performs: it generates the necessary system critical information needed for compiling the rest of the program. This information is stored in the form of a header file which is used only during the course of the program's compilation.

The configuration program generates the following files after execution:

| | |
|---|---|
| Etc/CHECKSUM | The checksums of all the system critical files. |
| Etc/environment | The detected configuration "environment" |
| Etc/custom.h | Special header file created to assist in compilation of software. |

The configuration program is completely stand alone and does not need to generate a visual report. Most of the security programs built into HostCHECK will not function until the configuration utility is executed for the first time.

111

HostCHECK Functional Des    .on
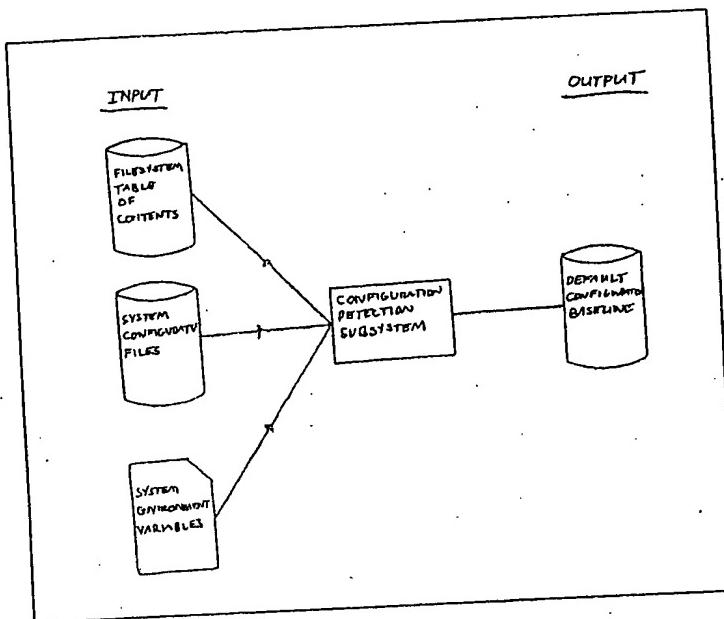


## Directory Scanner

The Directory Scanner looks for security flaws that develop in the file-system of the computer over a period of time and detects if "security critical" files have been tampered with.

Using ANSI C instead of scripts, the time it takes to search an entire file-system usually rangs from three to four minutes. In this time, a typical UNIX computer may have 30,000 to 50,000 files examined. Security problems that are searched for are:

- Globally read/writable directories
- Executable files that are globally modifiable
- Globally writable setuid/setgid files
- SUID files that have changed ownership
- SGID files that have changed group
- SUID/SGID/Critical files that have changed permission
- Newly created SUID/SGID files
- Protected files that have changed ownership
- Protected files that have changed group
- SUID/SGID/Protected files that have been deleted
- SUID/SGID/Protected files that have been tampered
- Incorrect device driver permissions
- Tampered device driver permissions
- Incorrect Device Ownership
- Incorrect Device Group

If any security problems are identified, the system administrator is prompted for a quick fix if the program is capable of providing one. All of the information associated with security problems and that were corrected are forwarded to the reporting system.

### Security Profiler

The Security Profiler security module examines individual file permissions for non-standard configuration. Files contained in a pre-made database are checked against files on the system. If the files on the computer differ from the files in the database, a suggestion is made to change the file's rights to the suggested rights.

The permissions are determined as follows:

1. Files selected are typically used UNIX files resisting in public binary executable directories (i.e. /bin, /usr/bin, /sbin, /usr/sbin, etc.) or common directories where insecurities may exist.
2. File permissions are revoked for regular users so that they are allowed to execute the file, but not read the contents or change the file.
3. Directories that have common insecurities (/var/spool/crontab, /usr/spool/mail, etc.) are set with proper secure permissions.
4. Other files that are commonly misconfigured are added as well (/etc/shadow, /etc/master.passwd, /etc/profile, /, etc.)
5. Any other changes follow the general rule "Less permission is better" without shutting services off for the regular user.

### Password Cracker

The Password Cracker security module examines the DES encrypted passwords associated with each user for insecure password choices. The administrator can use this tool to test the strength of the system's front-end security. It is a well-known fact that many systems become compromised because users pick insecure passwords.

There are several functions of the Password Cracking suite that are both speed and functionality related. They are:

- Use of the high-speed "fcrypt" program by Eric Young, entered into the public domain.
- The use of "same salting", so that there will only be a single "salt" attempt per entire dictionary. This optimizes the speed of the security test.
- The integration of "Similar Salts", a technique that puts a ceiling on the amount of time necessary to examine the passwords of large numbers of users. If a computer has over a thousand accounts, then performance may improve 25-45%.
- Filtering of words to generate "pseudo"-words, such as replacing "t" with "+", making words like "tomato" into "+oma+o". There are 8192 filter combinations in the filtering mechanism.
- Create "GECOs" password guessing, to determine the technique used by the system administrator(s) of the computer to give out "never before used" accounts.
- Generation of "large", non-repetitive dictionaries so that multiple dictionaries can be used for testing that does not contain duplicate words.

113

PASSWORD FILE OR RESUME FILE

WORD LIST

PASSWORD CRACKER

LIST OF INSECURE PASSWORDS

RESUME FILE (IF APPLICABLE)
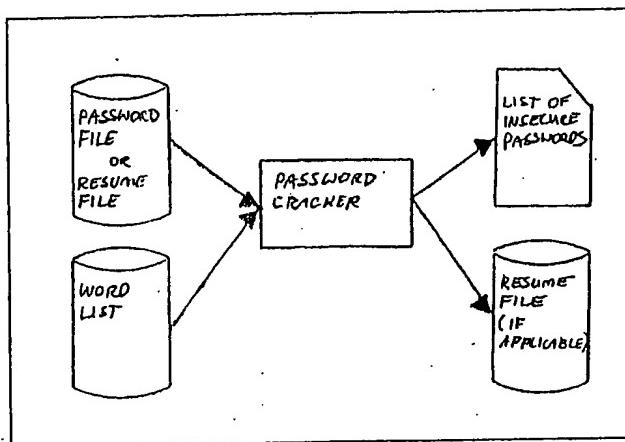
## Integrity Checker

Despite good management, many computer security problems exist on computers by default. A collection of these vulnerabilities is stored in the HostCHECK's database and checks are made in order to find the holes on the local system. If a vulnerability is detected, the system administrator is notified of the problem as well as where he/she can locate more information about the problem and patches.

The database of security holes is mostly comprised of CERT, ASSIST, ALERT, Bugtraq, and other commonly referenced security bulletins and discussion groups. The information gathered from those documents is used to generate the check needed to determine if there is a security problem. Key checks on a program to determine if a computer hole is exists are:

- The type of computer/operating system being used
- A specific string in file(s) where the security hole might exist
- The access privileges of the file
- The owner of the file
- The group of the file
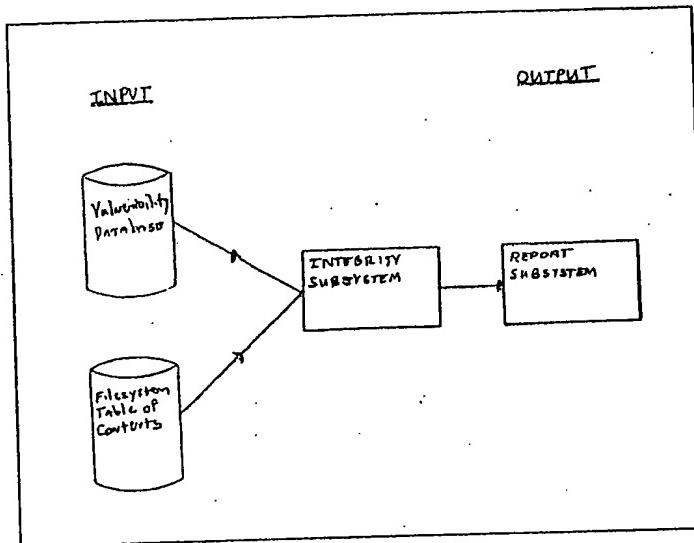
If ALL of the tests prove positive, then the Integrity Checker will report that there may be a security hole in the operating system. It will also report the filename, the nature of the security hole, and where the system administrator may locate additional information on the problem.

## User Manager

The User Manager is a dual-purpose system administration tool and computer security tool that oversees the user-level security on the system. It allows easy access to user account creation, easy creation of multiple groups, system wide searches for user account vulnerabilities and easy-to-use identification of logs relating to individual users.

There are different types of functions it performs, depending on the system configuration. The most commonly used features are:

- Creation of new user accounts
- Creation of new user groups
- Deleting and isolating files owned by a specific user
- Locating logs pertaining to an individual user
- Searching home directories for improper ownership
- Searching for non-existent home directories.
- Searching home directories for improper group
- Searching home directories for improper "+" in .rhosts file.
- Searching for insecure ".netrc" file

The User Manager is compatible with the traditional password/account generation methods as well as Shadow Password security package. The User Manager may or may not be capable of displaying information about recent user log-ins, depending on the operating system.

INPUT

OUTPUT

USERLIST

USERLIST

DEFAULT
USER
TEMPLATES

USER
SECURITY
SUBSYSTEM

DEFAULT
USER
TEMPLATES

FILESYSTEM
TABLE
OF
CONTENTS

FILESYSTEM
TABLE
OF
CONTENTS

REPORT
SUBSYSTEM
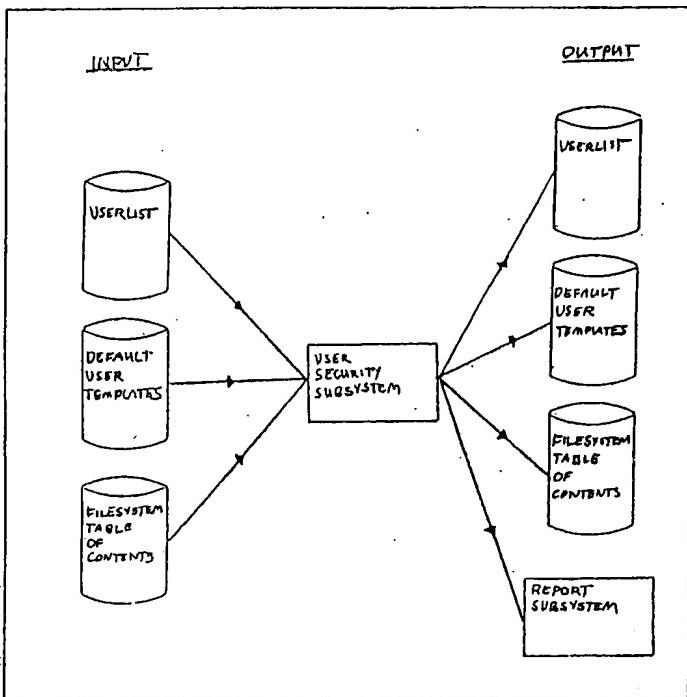
**Network Manager**

When computers are connected to a network, new vulnerabilities are opened up which an intruder can take advantage of. In order to keep the system secure, many of the networking security tasks are performed locally.

- Insecure configuration files
- Running of excessive system services
- Intrusion detection on TCP ports
- Checks for Promiscuous Mode

**Wipe**

Sometimes "sensitive" information appears on a computer when it shouldn't. Guarantees need to be made for deleted file contents to vanish from the computer permanently. To accomplish this, HostCHECK has a program that conforms to the Department of Defense guidelines for the destruction of the file's contents from the hard drive. This is accomplished by overwriting the file with bit patterns and text multiple times and verifying the information was changed.

The procedure used to securely remove a file is as follows:

Step     Description

One      the file is overwritten by a bit pattern such as 0101.

Two     The file-system is synchronized in order to force data to be written to the drive.
Three   the file is read again in order to verify that the file has been overwritten with the 0101 pattern.
Four    Steps one, two, and three are repeated with different bit patterns for a total of three different
        patterns.
Four    The link to the file is removed.

This follows the directive set down by the Department of Defense standards for removal of sensitive
information, as defined by the DATA REMNANCE GUIDE (NCSC-TG-025). Here is the section
describing the technique for removal which the Wipe utility was uses:

### 5.1.1    OVERWRITING

Overwriting is a process whereby unclassified data are written to storage
locations that previously held sensitive data. To satisfy the DoD clearing
requirement, it is sufficient to write any character to all data locations in
question. To purge the AIS storage media, the DoD requires overwriting with a
pattern, then its complement, and finally with another pattern; e.g, overwrite
first with 011 0101, followed by 1100 1010, and then 1001 0111. The number
of times an overwrite must be accomplished depends on the storage media,
sometimes on its sensitivity, and sometimes on differing DoD component
requirements. In any case, a purge is not complete until a final overwrite is
made using unclassified data.

## Subsystem Diagrams

## Setup Software Subsystem

HostCHECK's installation program is used to initially install and configure the package as a whole. This
tool assists in walking through the setting up of the security suite so that the maximum protection will be
given to the computer system while requiring a minimum amount of knowledge about how HostCHECK
works.

This package performs a series of steps needed to baseline the host. Although all of these actions can be
performed from the command line, these steps are automated by this program for convenience sake. The
steps that are performed by the setup program are:

1.    The Configuration program is run first to identify the current configuration of the system,
      identifying the important aspects of the system's security that the Directory Scanner
      won't be able to identify by itself. It also detects standard features of the computer that
      need to be used for special consideration as to the functioning of the security modules.

2.    Install asks for parameters to special options, such as how to print to the printer, who to
      send electronic mail reports to, and report characteristics such as width, margins, and
      height. These are saved in the etc/configuration file.

3.    The Directory Scanner is run, merging the information it detects with the information
      provided by the Configuration tool. When run by the setup software, no security
      information is displayed when the Directory Scanner runs. This is to avoid confusing the
      installer.

4.    The profiler is then called upon to establish a baseline for the security of approximately
      1000 files on the host. This is saved into a database and is used to make certain that the
      security of these files don't change.

## Mailer Software Subsystem

The Mailer Program handles communication between the systems and subsystems, usually directing information to the reporting system but can also be used to send reports to the printer, e-mail, or to the screen.

## Composite Report Generator

The composite report generator is a handler that allows the execution of multiple tests in a single running. Unlike the tests run individually, these tests are not supposed to have human interaction. These tests all have similar traits:

- Autocorrection is turned off
- Execution of tests occurs after the previous test finishes.
- Tests are run using the "standard" configuration by default.

## Mark/Unmark Subsystem

The Mark/Unmark utility assists in determining if a file is critical. If a file is critical, but the Configuration and/or Directory Scanner program could not determine the critical nature of the file (e.g., the system administrator installed application software online that needs to be protected), then the Mark utility is required to secure the file. Marking a file with the Mark utility will cause that file to become protected by the Directory Scanning utility.

Unmarking a file will cause a file to be removed from the list of critical files maintained by the Directory Scanner. However, if the file is a critical file by default, then the unmarking utility will cause the file to be removed, where it will be re-added to the security list next time the Directory Scanner is run.

118

## Password Checker

1. Password file is loaded, sorted by the salt field
2. Next (or first) password entry is loaded, including all similar salted line
3. User(s) are checked for non-existant passwords
4. User(s) are checked for passwords same as their user name
5. User(s) are checked for passwords guessable from their GECOs field
6. A dictionary file is opened, and the pointer is moved to the first word in the list
7. 5000 words are loaded from the dictionary into memory
8. The pointer is moved to the first word in the loaded list of 5000 words
9. If filters are used, apply the filter to the word to form a new wordlist
10. The pointer is moved to the first word in the new wordlist
11. The word is encrypted and salted via DES
12. If the encrypted entries in the password field are the same as the guessed password, the user guessed is removed from the list
13. Advance pointer. If pointer is not at the end of the list, go to step 11
14. Deallocate memory used in the current list
15. Advance dictionary pointer. If pointer is not at the end of the file, go to step 7
16. If not at end of user list, go to step 2
17. Save report
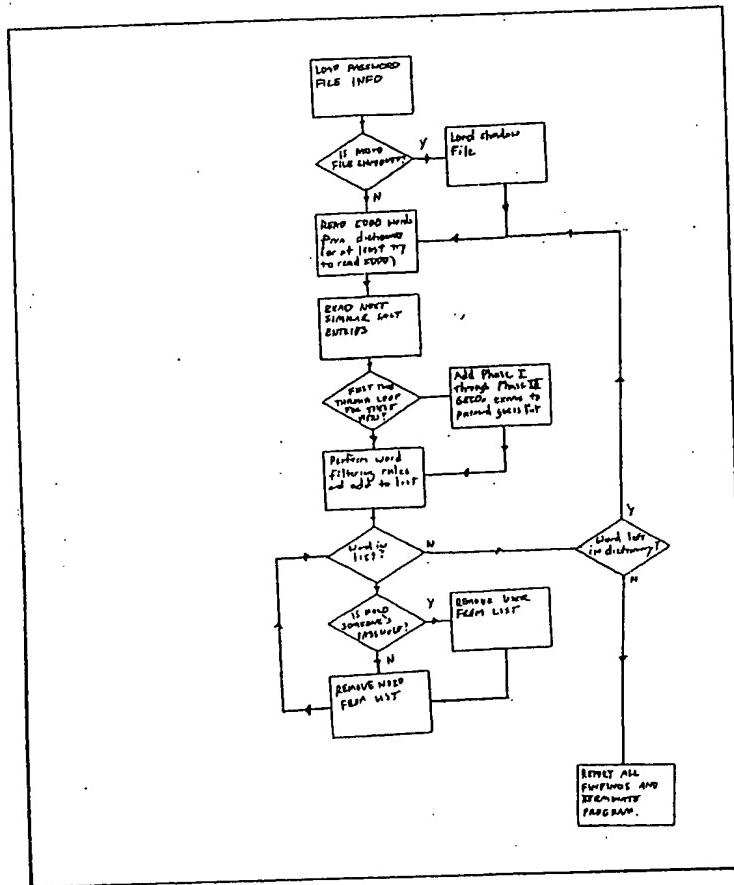
If the software is interrupted by a "hangup" signal, the following takes place:

1. Current userlist is saved in the file ./spool/passchk/saved
2. Report is saved

The "saved" file is a standard password file containing all of the password entries not yet examined. Restarting the Password Checker selecting that file as the password file will resume the cracking process at the last given point.

## Secure File Wipe

The Wipe utility takes a filename from the command line, and attempts to remove it securely without leaving traces of the file's existence behind. If the file passed to the Wipe utility isn't a direct filename (i.e., it contains ".." or "." as a path, possibly to fool the system into wiping out a device instead of a file, the Wipe utility will not erase that file.

The procedure used to securely remove a file is as follows:

1. The file is overwritten by a bit pattern such as 0101.
2. The file-system is synchronized in order to force data to be written to the drive.
3. The file is read again in order to verify that the file has been overwritten with the 0101 pattern.
4. Steps 1, 2, and 3 are repeated with different bit patterns for a total of three different patterns.
5. The file is then overwritten with the text "The quick brown fox jumps over the lazy dog" in order to simulate "non-sensitive" information.
6. The link to the file is then removed.

126
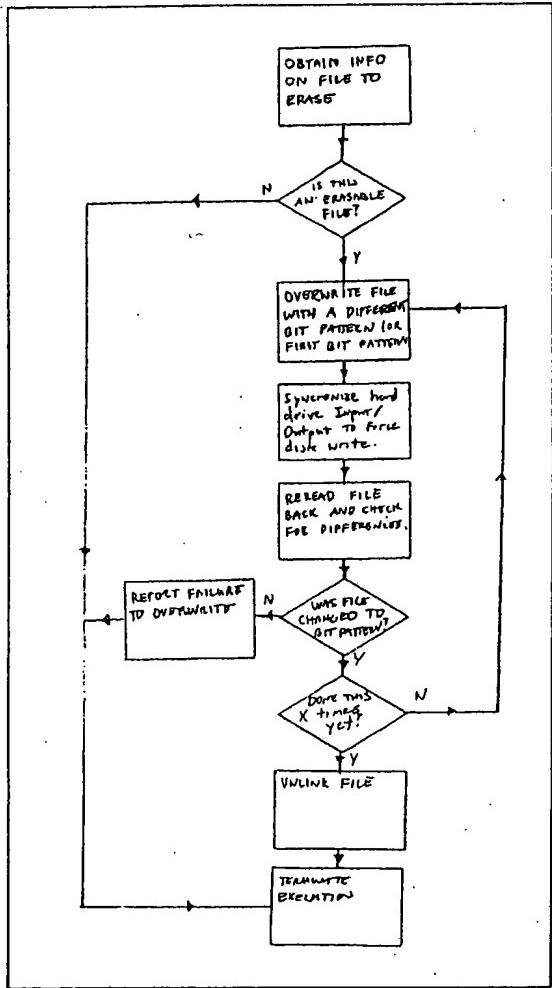
## HostCHECK Menu System

The HostCHECK Menu system exists in two forms, the Graphical User Interface version, which allows the program to use a graphical, event-driven interface; and the Text version, which allows people using the computer via a dial-up adapter or from a dumb terminal to be able to control the software package. Running the Menu System with a –X in the command line will force the GUI to be used, otherwise the software will default to text mode.

GUI Menu System
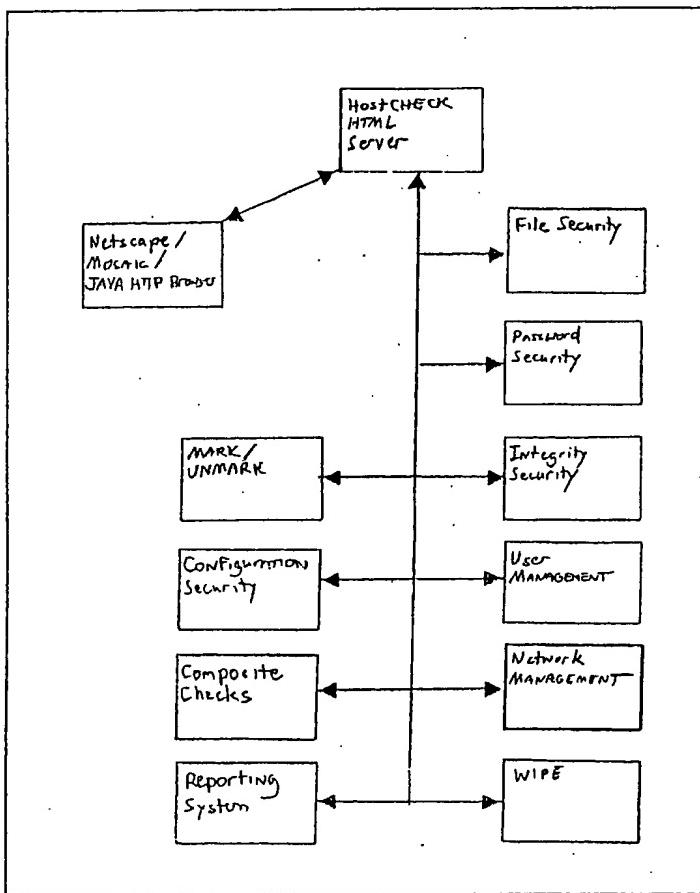
Text Menu System

The Text Menu has limitations on what it can and cannot do, as a result of its inability to display graphical text. It cannot display information in color, nor can it display a fancy HTML report. The reports that are generated are ASCII based, but contain the same information the HTML reports do.

The structure of the menu tree is:

COMPOSITE SECURITY CHECKER
        Full Security Audit with Automatic Correction
        Full Security Audit with Manual Correction
        Full Security Audit without Corrections
        Quit/Exit Full Audit Menu

128

built in order to handle all keyboard based I/O so that a degree of control can be maintained over what the user of the application types.

- readline() is a function used to read a string of text from the command line. It can display the text being entered in highlighted form, it can limit the amount of information that goes into the string by performing bounds checking, and it can even set restrictions on quantity of numeric values entered in the string.
- getkey() allows a single keystroke to be captured, without displaying the data to the screen, and relinquishes control back to the program logic immediately after the keystroke is pressed.
- YesNo() prompts the user to answer "Y" or "y" for Yes, "N" or "n" for No. Returns 1 if they answered yes, 0 if they answered no.
- common_prompt() is a system of prompting used to communicate with the GUI, if there is one currently running, or with the text mode interface. This prompting system handles all of the standard prompting for all of the HostCHECK utilities.

### General Common Functions

The general functions are just translations for very common functions that are needed for common usage, which may take place throughout the code's logic, and very well may be just overlooked by the library functions as things which needed to exist in general programming usage. There are two functions currently in this genre.

- shadowdate() is a buggy, "yet serves its purpose" system for determining dates from shadow password file entries.
- itoa() is an "integer to ASCII string" conversion system.

### Rhost File Identification

This function is used primarily for identifying a "+" in the .rhost file possessed by a user. However, it also can be used by other subsystems for detecting a "+" in files such as "hosts.equiv", etc. The check_rhost_file() function receives a filename, and returns positive if the file contains a hazardous "+" symbol.

### Syncronization Subsystem

Once a program is finished, in order to facilitate communication between the different software packages which may be currently running, a call is made to the singular syncronize() function. At the moment, this program merely runs the ./bin/mailer program and allows it to process any new information that has been generated.

## Data Models

- Data Files
- Database tables
- Data structures
- Data Dictionaries

## Data Files

### Profile Database

The file baselining is performed in the profile database file, and is stored in the form of a comma and quotes delimited ASCII database. All of these fields are of type VARCHAR, and can be up to XX kilobytes in size. This file is automatically generated and updated by the File Security Subsystem.

135

- Directory of File
- Filename
- Permissions
- Owner
- Group
- Date
- File Size
- Checksum
- SHA-1 Secure Hash

## Vulnerability Database

The vulnerability database is very similar to the profiling database in that it tries to describe files in which a vulnerability may be present, without attempting to look for an exact match. This is because the checksum and/or SHA-1 secure hash may be different during each software release's compilation. The file is stored in the form of a comma and quotes delimited ASCII database. All of these fields are of type VARCHAR, and can be up to XX kilobytes in size.

- Bug ID #
- Operating System
- Directory
- Filename
- Setuid?
- Setgid?
- Owner
- Group
- Permissions
- Date of Patch
- URL of Patch

## Report Database

All of the reports generated by HostCHECK come from a canned report database. This database, comprised primarily of paragraphs, has two fields. The file is stored in the form of a comma and quotes delimited ASCII database. All of these fields are of type VARCHAR, and can be up to XX kilobytes in size.

- Name of Text Segment
- Text of segment, in paragraph form.

## Recovery Database

Before the actual file baselining is performed, a snapshot of the security of many of the system's critical files is performed in order to allow some files to have their security recovered in case of error. The recovery file is stored in the form of a comma and quotes delimited ASCII database in the same fashion as the standard Profiling Database. All of these fields are of type VARCHAR, and can be up to XX kilobytes in size. This file is automatically generated and updated by the File Security Subsystem.

- Directory of File
- Filename
- Permissions
- Owner
- Group
- Date          (unused)
- File Size     (unused)

136

## Mark/Unmark Subsystem

The Mark/Unmark utility assists in determining if a file is critical. If a file is critical, but the Configuration and/or Directory Scanner program could not determine the critical nature of the file (e.g., the system administrator installed application software online that needs to be protected), then the Mark utility is required to secure the file. Marking a file with the Mark utility will cause that file to become protected by the Directory Scanning utility.

Unmarking a file will cause a file to be removed from the list of critical files maintained by the Directory Scanner. If the file is a critical file by default, however, then the unmarking utility will cause the file to be removed, where it will be re-added to the security list next time the Directory Scanner is run.

# APPENDIX A - ALPHABETIZED FUNCTION LIST

| Function | Header | Description |
|---|---|---|
| accpol | account.h | Tests DoD Accounting Policy Criteria |
| add_to_header | format.c | Adds text to header segment |
| add_to_segment | format.c | Adds text to a segment |
| add_user | adduser.h | Creates a New User from User Entry |
| append_facts | facts.h | Merges FACTS to end of SIF file |
| archive_report | mailer.c | Copies generated report into system archive |
| ascii_center | visual.h | Centers string regardless of display type |
| ascii_text_wrap | visual.h | Wraps text regardless of the display type |
| asspol | assure.h | Tests DoD Assurance Policy Criteria |
| bold | visual.h | Displays text entered as bold face |
| bracketcees | filter.c | Changes letter C to the symbol [ |
| center_text | visual.h | Centers string and displays on current line |
| certification | hostguard.c | DoD Certification Menu |
| changeprompt | dirscan.c | Displays prompt asking if file's permissions should be changed |
| check_device_permissions | devices.h | Compares device permissions |
| check_device_permissions | devices.c | True if device permissions are normal |
| check_rhost_file | rhost.h | Checks for + in rhost file |
| checksum_everything | dirscan.c | Launches checksumming process |
| clean_groups | groups.h | Properly unallocates GROUPS structure |
| clean_passwd_list | passwd.h | Cleans password structure in correct manner |
| clear_screen | visual.h | Clears the text screen |
| clear_space | visual.h | Clears a section of the screen |
| clearenv | environm.h | Removes environment variable from list |
| close_database | database.h | Closes an open database |
| compare_files | sums.c | Returns true if both files are identical |
| compute_checksum | sums.h | Computers File Checksum, Length, and SHA-1 |
| configure | hostguard.c | System Configuration Menu |
| create_new_group | grman.c | Creates a new psuedo group |
| curleyC | filter.c | Changes letter C to the symbol { |
| cursor_hide | visual.h | Hides the cursor |
| cursor_normal | visual.h | Restores cursor to normal |
| cursor_standout | visual.h | Bolds the cursor |
| deluser | deluser.c | Deletes user from system |
| detect_filesystem | filesys.h | Guesses type of filesystem |
| dialogwinon | screen.h | Displays a Text Dialog Box |
| directory_scanner | hostguard.c | Directory Scanner Menu |
| dirscan_report | Reporter dirscan.c | Generates a Dirscan report from recent data |
| display_group | grman.c | Displays group list |
| display_group_menu | grpmenu.c | Enters group manager/menu system |
| display_inetd | netman.c | Displays inetd entry |
| display_inetd_menu | inetmenu.c | Displays inetd configuration |
| display_menu | menu.h | Displays text menu |
| display_user | user.h | Displays a single user |

| Function | Header | Description |
|---|---|---|
| display_user_menu | usermenu.c | Displays user menu |
| dollarS | filter.c | Changes letter S to the symbol $ |
| edit_user | user.h | Edits a single user |
| elles | filter.c | Changes letter L to the symbol 1 |
| email_file | mailer.c | Mails file to system administrator |
| evaluate | config.c | "Evaluates" a simple script |
| examine_home_dir.. | homedir.h | Examines user for security problems |
| exclamation | filter.c | Adds exclamation point to words in word list |
| fact2string | facts.h | Converts a FACTS structure to a SIF line |
| fileitem | user.h | Displays items of file list |
| filter | filter.h | Generates permutations of a single word |
| filtered_check | hostguard.c | Launches Password Checker (w/filtering) |
| find_dicts | dicts.h | Locates dictionaries for password cracker |
| find_ownership | owner.h | Searches for files owned by a single user |
| finish_report | report.h | Generates a Directory Scanner final SIF file |
| free_word_list | filter.c | Cleans words from word list in proper manner |
| full_automatic | hostguard.c | Full Examination w/automatic repairs |
| full_manual | hostguard.c | Full Examination w/manual repairs |
| full_none | hostguard.c | Full Examination w/no repairs |
| full_report | hostguard.c | Composite Examination Menu |
| gecos_check | hostguard.c | Launches Password Checker (only GECOs) |
| genenv | config.c | Generates a new ENV entry |
| generate_file_table | Reporter dirscan.c | Generates a table of file data from FACTS |
| generate_gecos | gecos.h | Generates word permutations from GECOs |
| generate_group_table | Reporter dirscan.c | Generates a table of group data from FACTS |
| generate_table | Reporter dirscan.c | Generates a table of data from FACTS |
| generate_user_table | Reporter dirscan.c | Generates a table of user data from FACTS |
| genkey | genkey.h | Generates a Copy Protection Key |
| get_file_rights | files.h | Obtains file's permissions |
| getkey | cmdline.h | Reads a single character from input |
| googleO | filter.c | Changes letter O to the symbol @ |
| gotoxy | visual.h | Moves cursor to screen location X Y |
| group_manager | grman.c | Enters group editing system |
| homedir_security | homedir.h | Checks general home directory permissions |
| ifenv | envirornm.h | Checks to see if environment variable is set |
| incritical | dirscan.c | Returns true if file has been flagged critical |
| inetd_configuration | netman.c | Displays inetd configuration |
| inetd_filter | inetd.h | Standardizes inetd entry |
| initialize_screen | visual.h | Initializes terminal display |
| install | install.h | Installs HostCHECK software |
| installation | hostguard.c | Launches Installation Program |
| integ_report | Reporter integ.c | Displays a report of vulnerabilities |
| integrity_checker | hostguard.c | Integrity Checker Menu |
| interrupt_handler | cmdline.c | Basic Interrupt Handler Function |
| inverse | visual.h | Displays text as Inverse |

---

**DMW Worldwide Proprietary Information**

| | |
|---|---|
| certify.tes* | DoD Certification Test |
| mail.tes* | Mailer System Test Script |
| solaris2.aft* | (?) |
| solaris2.opt* | (?) |
| solaris2.rel* | (?) |

tools:
| | |
|---|---|
| 4wl.c* | 4 Weak Links |
| keys.c* | Keystroke Logging/Displaying |
| makefile* | Makefile for weak.c |
| weak.c* | Weak Links Prototype |

unmark:
| | |
|---|---|
| clean* | Removes object files and executables |
| makefile* | Makefile for Unmark |
| unmark.c* | Unmark Program |

userman:
| | |
|---|---|
| adduser.c* | Routines to create a new user |
| clean* | Removes object files and executables |
| deluser.c* | Routines to delete a user |
| grman.c* | Group Manager Routines |
| groups.c* | Group handling routines |
| grpmenu.c* | Menu for Group Editing |
| homedir.c* | Home Directory Permissions |
| lastlog.c* | Last Log Examination Functions |
| makefile* | Makefile for User Manager |
| owner.c* | Functions to locate file ownership |
| skel.c* | Functions to handle Skeleton Directories |
| user.c* | User handling routines |
| userman.c* | User Manager Main Routine |
| usermenu.c* | User Editing Menu |

wipe:
| | |
|---|---|
| clean* | Removes object files and executables |
| filesys.c* | Link to File System Library |
| makefile* | Makefile for Wipe |
| wipe.bac* | Backup file for Wipe (NOT Wipe data) |
| wipe.c* | Wipe Utility |

wipe.old:
| | |
|---|---|
| clean* | Removes object files and executables |
| files.c* | Link to Files Library |
| filesys.c* | Link to File System Library |
| makefile* | Make file for Linux Wipe |
| wipe.c* | Linux Wipe |

wish:
| | |
|---|---|
| wish.hpu* | Precompiled Windows Shell for HPUX |
| wish.lin* | Precompiled Windows Shell for Linux |
| wish.sol* | Precompiled Windows Shell for Solaris |

xwindows:
| | |
|---|---|
| audit.tcl* | Security Audit TCL |
| comp.tcl* | Composite Check TCL |
| config.tcl* | Configuration TCL |
| confirm.tcl* | Confirmation (Yes or No) TCL |

---

**DMW Worldwide Proprietary Information**